Windows в Linux » Google » OOo » Mono

LXF DVD

8 Гб отборных дистрибутивов!

Плюс: Opera 9.2, Blender, Ответы, Thunderbird и многое другое!

LINUX FORMAT

Главное в мире Linux

Июль 2007 № 7 (94)

Конкурс дистрибутивов

» Безопасность » Производительность
» Поддержка » Ассортимент пакетов
» Сообщество

Как выступит ваш любимый вариант Linux?



Жизнь рядом с Windows

Запускайте Windows-приложения в Linux и наоборот: кросс-платформенная нирвана начинается здесь – на с. 40

Офис от Google: время выйти в Сеть? с. 48

Web-чертеж

Изучите тэг canvas и рисуйте с JavaScript с. 78

Wireshark

Анализ сетевого трафика – для пользы и для забавы! с. 70



“Станете ли вы рисковать своим делом и строить бизнес исключительно на Open Source?”

Андреас Типальдос из Xandros – бизнесмен от Linux

К Вашим услугам...

Команда LXF снова с вами! В этом месяце они тестировали лучшие дистрибутивы Linux, поэтому мы спросили: каков из себя идеальный дистрибутив, который никогда не был выпущен?



Пол Хадсон
GNU/X11/Gnome/OOo/Vim/Gimp/Mono/Evolution/Linux – понравится всем, кроме отдела маркетинга.



Грэм Моррисон
У Limerix не будет пингвина
Надоел он им хуже хинина.
Зато будет червяк
(Может, автор – рыбак?),
Хоть и странная выйдет картина.



Майк Сондерс
Snookbuntu. Разновидность Ubuntu, где щелчок мышью звучит как удар шаром о борт, а единственное запущенное приложение – Foobilliard.



Эфрейн Эрнандес-Мендоса
Дистрибутив, показывающий при старте видеозапись голов, забитых сборной Мексики на Кубке Мира – всех четырех часов.



Ребекка Смолли
Встраиваемый дистрибутив, который работал бы на моем калькуляторе. Это единственное электронное устройство, с которым я чувствую себя комфортно.



Ладислав Боднар
Slackware 1.0 (1993). С ядром 0.99pl11-alpha и 24 дискетами первосортного ПО, компьютерная революция уверенно двинулась по намеченному пути.



Нейл Ботвик
RTFM Linux: дистрибутив, находящий ответы на любой вопрос и читающий документацию, пока вы заняты своими делами.



Энди Хадсон
Дистрибутив «на замену Windows»: сохраняющий все внешние признаки этой ОС, но имеющий ядро Linux внутри.



Энди Ченнел
Хочу постмодернистский дистрибутив, рефлексивно оспаривающий все пополнозновения на тезис «Правды нет».



Дистрибутивы на ринге

Именно так назывался спецрепортаж этого номера до тех пор, пока не прошел через добрые руки литературного редактора. Быть может, в этом действительно что-то есть? Конкурс (в особенности конкурс красоты) — это, конечно, не всегда дружеские посиделки (случается, даже бритвы в туфли соперницам подкладывают), но по крайней мере, мероприятие, не агрессивное по своей природе: претенденты просто показывают себя с лучшей стороны, а жюри выставляет оценки. В среде пользователей Linux, однако, зачастую применяется иной подход к определению победителя: думаю, с «религиозными войнами» встречался каждый, кто хоть раз заходил на форум Open Source-тематики. Зачастую весь спор строится на переинчивании аргументов соперника: «А Mandriva лучше, потому что ее можно настраивать из графики. — GUI для настройки? Да я за это...». Приятно, конечно, что такой разговор можно продолжать до бесконечности (этим он напоминает старую игру в «физиков и лириков», которая заканчивается, когда «мама позовет ужинать»), но ведь можно поразить оппонента не словом, а делом. Ругаете Slackware за отсутствие приложений и поддержки чего-то интересного, но при этом соглашаетесь, что это простой и быстрый дистрибутив? Соберите необходимые пакеты и отправьте на LinuxPackages.net. Недолюбливаете OpenSUSE за излишний вес? Присоединяйтесь к проекту OpenSUSE SLICK (сейчас, кстати, самое время это сделать). Тогда и разговор станет более предметным, и пользы для сообщества будет больше. **LXF**

Валентин Синицын » Главный редактор info@linuxformat.ru

Миссия журнала

- Пропаганда свободного ПО в России
- Продвижение решений с открытым кодом в бизнес-сообществе
- Поддержка российского Open Source сообщества
- Организация трибуны для разработчиков свободного ПО
- Обратная связь между разработчиками и потребителями ПО

Corporate Patron
Free Software Foundation

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Web-сайт: www.linuxformat.ru
› **Адрес редакции:** Россия, Санкт-Петербург, ул. Гончарная, 23, офис 54.
› **Телефон редакции:** (812) 717-00-37. Дополнительная информация на стр.128

Содержание

Июль 2007 № 7 (94)

Весь номер – прямо как на ладони: приятного чтения!

Учебники

LXF DVD94

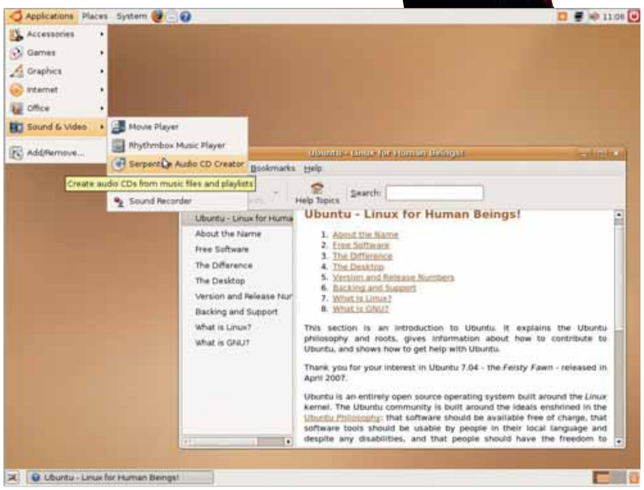


Ubuntu
Де-факто стандарт для домашних пользователей, отличающийся от стандартной версии настолько, насколько это возможно – мы добавили сотни дополнительных пакетов. Счастливые пользователи!

Mandriva 2007 Spring
Ищете легкий путь в мир Linux? Этот дистрибутив полон прекрасных программ и великолепных графических конфигураторов.

Debian 4.0
Крепкий, рафинированный, надежный дистрибутив – идеальный выбор для рабочих станций и серверов опытных пользователей.

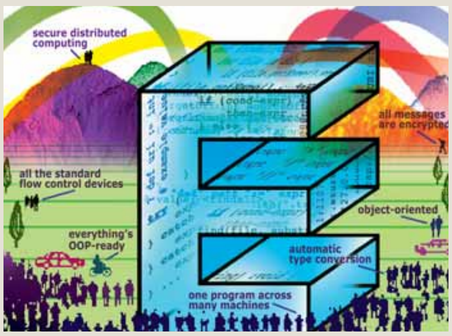
SimplyMepis
Один из самых привлекательных Live-дистрибутивов, хороший способ проверить Linux-совместимость вашего нового ПК.



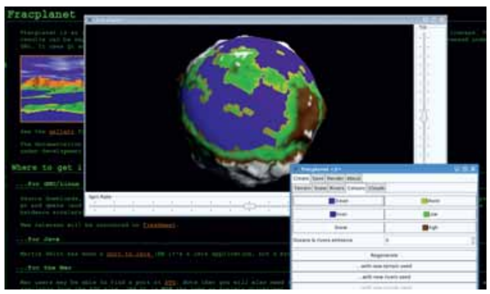
› Да! Ubuntu 7.04 есть на нашем диске. Если коричневый — не ваш цвет, попробуйте три других дистрибутива.

Что за штука...

E? Подсказка: это не C и не D... с. 30



LXF HotPicks



› Когда этот мир вам надоест, сотворите себе новый.



LXF ИНТЕРВЬЮ

«Станете ли вы рисковать своим делом и строить бизнес исключительно на Open Source?»

Андреас Типальдос с. 26



Кросс-платформенная нирвана с. 40

Спецрепортаж из двух частей: взгляните на свою ОС в новом свете



Подпишись на Linux Format и сэкономь!



Содержание

LXF DVD внутри! с. 125

Спецрепортаж

Конкурс дистрибутивов

Кто выйдет из этой битвы победителем? с. 32

А также...



Постоянные рубрики



› **OpenGL не дает погонять на пингвинах? Гони на стр. 104!**

Обзоры



› **Обзор новой Linux-версии *Dofus*, выпущенной для английского рынка.**

Сравнение: онлайн-резервирование



LINUX FORMAT Новости

ГЛАВНЫЕ НОВОСТИ: » Mandriva приходит в Россию » GPL3 » Microsoft в каждый дом » FOSS от ЮНЕСКО » И снова об ODF » Linux-мобилизация

» Рубрику ведет Илья Шпаньков



Mandriva получает прописку в России

Последние дни весны станут для российского Linux-сообщества памятной датой – 28 мая компания Mandriva открыла в Санкт-Петербурге свое представительство и центр разработки. Таким образом, в создании одного из самых популярных в мире дистрибутивов теперь будут принимать участие не только французские и бразильские разработчики, но и российские программисты. Стоит отметить, что новый филиал пришел в Россию не с пустыми руками: одновременно с официальным объявлением о новом представительстве широкой публике были предложены и специализированные версии дистрибутивов Mandriva 2007 Spring, адаптированные для использования в Российской Федерации. Отличие от оригинальной версии заключается в использовании таких компонентов, как *OpenOffice.org* в сборке «Инфра-ресурса», а также пакета библиотек отечественной компании Etersoft, позволяющих выполнять в среде Linux такие популярные на российских предприятиях приложения, как *1С:Предприятие 7.7-8.1*, *Инфо-Бухгалтер 8.X*, *Гарант F1* и *Гарант F1 Турбо*, *Консультант+ 3200/3300*, *Кодекс 5.2*, *ABBY FineReader 7.0-8.0*, *ABBY Lingvo 9.0*, *КОМПАС 2D* и другие. Двумя неделями позже был анонсирован продукт Mandriva Corporate Desktop 4.0, ориентированный на применение в бизнесе.

Примечательно, что серьезность намерений компании Mandriva в России подтверждает и наличие учебных центров, расположенных пока в Москве и Санкт-Петербурге, а в будущем планируемых к открытию и в других крупных российских городах. Спектр программ обучения охватывает все уровни, начиная от начального знакомства с Linux-системами, и заканчивая сетевым администрированием. Стоит также отметить, что все курсы сертифицированы независимой организацией LPI (Linux Professional Institute) и не имеют жесткой привязки к конкретному дистрибутиву Linux, что позволяет значительно расширить число желающих получить соответствующие навыки.

Подобные шаги разработчика Linux-дистрибутивов с мировым именем также говорят о том, что сегодня российский рынок ПО становится одним из наиболее перспективных для продвижения свободного программного обеспечения и немалую роль в этом сыграла борьба российских правоохранительных органов с распространением нелицензионного ПО, а также повышение компанией Microsoft цен на свои программные продукты в России. До последнего времени потребности российского рынка в операционных системах Linux обеспечивались в основном двумя российско-украинскими фирмами – AltLinux и ASPLinux. Теперь у них появился достаточно сильный конкурент, способный значительно потеснить позиции вышеназванных компаний. Впрочем, в большей степени компания Mandriva нацелена на пользователей Windows-систем, не желающих или не имеющих возможности переходить на новые дорогостоящие и весьма требовательные к аппаратным средствам разработки редмондского гиганта, поэтому можно достаточно уверенно прогнозировать уменьшение числа приверженцев проприетарного ПО и рост аудитории Linux-пользователей, при этом активные маркетинговые действия французско-бразильско-российской компании могут косвенно помочь и отечественным разработчикам, не обладающим аналогичными финансовыми возможностями.

mandriva.ru

Кстати



9 июня 2007 года специалисты Mandriva.Ru провели в городе Ханты-Мансийске мастер класс по технологиям Linux, который начальным этапом продвижения свободного ПО на территорию Ханты-Мансийского автономного округа – Югры. Семинар встретил восторженные отклики представителей бизнеса и государственных структур из Ханты-Мансийска, Сургута, Нижневартовска и других городов Уральского федерального округа. По его результатам было принято решение об открытии центра компетенции Linux в Ханты-Мансийске, а со временем — и в других городах.

Кто боится GPL v3

В последний день весны был опубликован заключительный, четвертый по счету черновик текста лицензии *GPL v3*. До выхода окончательной редакции остались считанные недели и события, хоть как-то связанные с обновлением открытой лицензии, показали, что ее релиз может значительно изменить позиции различных компаний на рынке. Прежде всего, Ричард Столлмэн [Richard Stallman] опубликовал статью, в которой постарался еще раз пояснить основные пункты новой редакции *GPL* и убедить разработчиков использовать именно этот вариант для своих свободных программ. В частности, особое внимание он уделил пунктам, касающимся предотвращения в будущем заключения договоров, аналогичных соглашению между Microsoft и Novell. Как оказалось, его внимание к данной проблеме было отнюдь не чрезмерным: за последнее время аналогичные договоры заключили с редмондским гигантом такие известные на Западе поставщики Linux, как Xandros и Linspire. Обе компании поспешили успокоить окружающих, что данное сотрудничество совсем не означает признания нарушения Linux-разработчиками каких-либо патентов Microsoft, но при этом представители последней при каждом удобном случае не забывают призывать остальных разработчиков свободного ПО платить за используемые патенты. Как это сделали другие, «правильные» компании.

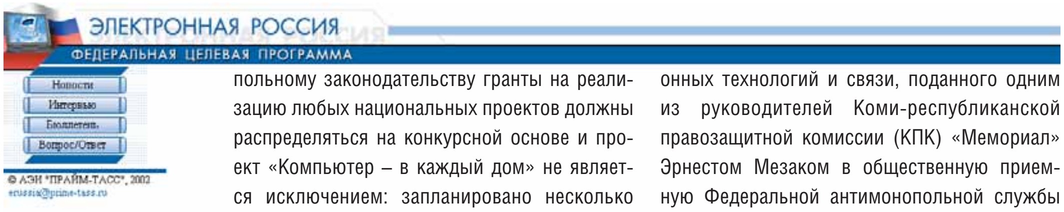
Другая острая тема, поднятая Столлмэном в свей статье, касается так называемой TiVo-изации, названной так в честь компании TiVo, использующей свободное ПО в своих решениях. Выход новой версии *GPL* и перевод под ее защиту свободных разработок позволит предотвратить использование свободного кода совместно с программами, обеспечивающими защиту DRM (Digital Restrictions Management) от взлома. Через несколько дней после публикации статьи Ричарда Столлмана представители TiVo сообщили прессе, что выход *GPL v3* может поставить под угрозу весь бизнес компании, одним из основных компонентов которого и является защита DRM. Но на самом деле ее участь выглядит не так пессимистично, как будущее Novell. В данный момент бизнес-модель последней оказалась в полной зависимости от действий Microsoft: если гиганту индустрии не захочется далее продолжать совместный бизнес с Linux-компанией в связи с неприятием *GPL v3*, то Novell потеряет практически все – в данный момент все основные продажи компании осуществляются благодаря заключенному договору.

Впрочем, Novell слишком сильный игрок, чтобы просто сидеть и ждать своей участи. Желая заранее раз и навсегда исключить возможные проблемы, связанные с патентами, как для себя, так и для других компаний, Novell совместно с фондом Electronic Frontier Foundation приступили к подготовке реформирования патентной системы, ставящей в последнее время все больше преград на пути научно-технического прогресса. Среди основных целей нового союза – активное лоббирование интересов национальных и международных организаций, занимающихся разработкой и принятием законов, позволяющих добиться значительных изменений существующей патентной системы. Особое внимание компании собираются уделить взаимодействию с участниками World Intellectual Property Organization (WIPO).

gplv3.fsf.org



Проект другой, вопросы те же



Не успел еще затихнуть скандал, связанный с предложениями Министерства информационных технологий и связи по обеспечению российских школ лицензионным ПО, как на повестке дня вновь подробности уже другого проекта, организованного ведомством Леонида Реймана. Речь идет о целевой программе «Компьютер – в каждый дом», стартовавшей 19 сентября 2006 года. Основная цель проекта – обеспечение компьютерной техникой жителей российской глубинки, оторванных от цифровых оазисов современных мегаполисов. Подобные целевые программы можно только приветствовать, но очень серьезные вопросы вызывает сама организация данного благого начинания. Дело в том, что среди организаторов проекта выступают не только госучреждения в лице Мининформсвязи и Федерального государственного унитарного предприятия «Почта России», но и московские представительства таких сугубо коммерческих компаний, как Microsoft и Intel. По российскому антимонопольному законодательству гранты на реализацию любых национальных проектов должны распределяться на конкурсной основе и проект «Компьютер – в каждый дом» не является исключением: запланировано несколько конкурсных мероприятий, которые должны выявить наиболее достойных кандидатов на роль поставщиков компьютерной техники или банковских организаций, предоставляющих кредиты на расходы в рамках проекта. И все бы ничего, но в технических спецификациях на компьютерную технику, поставляемую в сельские районы, фигурирует исключительно продукция компаний-организаторов: на долю Intel выпадает обеспечение процессорами, а Microsoft берет на себя программную часть проекта. Таким образом, данные компании получают монопольное право на распространение своих коммерческих продуктов в рамках российской национальной программы.

Данный факт стал причиной заявления о нарушении антимонопольного законодательства российским Министерством информационных технологий и связи, поданного одним из руководителей Коми-республиканской правозащитной комиссии (КПК) «Мемориал» Эрнестом Мезаком в общественную приемную Федеральной антимонопольной службы (ФАС). Впрочем, решение подобной проблемы вряд ли будет простым: если антимонопольный комитет признает подобный способ участия коммерческих компаний в реализации национальных проектов незаконным, то вся программа окажется на грани срыва в связи с потерей половины организаторов, а если оставить компаниям Microsoft и Intel прежние организаторские «должности», то вряд ли можно будет рассчитывать на то, что они не будут препятствовать продвижению продукции компаний-конкурентов. В любом случае, в будущем российским чиновникам придется более тщательно подходить к разработке национальных программ, чтобы не позволять отдельным коммерческим компаниям получать необоснованные преимущества перед другими участниками рынка.

Выбираем свободное ПО вместе

МОО ВПП ЮНЕСКО «Информация для всех» представил проект сборника прикладных программ «Свободное ПО для образования, науки и культуры», разработанный в рамках одноименного проекта. Публикация ориентировочного списка приложений осуществлена в целях получения от будущих пользователей и сторонних экспертов различных корректировок и пожеланий, которые позволят подготовить действительно полезный комплект свободных программ, предназначенных для работы в среде Windows. Следует отметить, что первый сборник свободного ПО в рамках данного проекта был выпущен еще в марте 2007 года силами LinuxCenter и завоевал достаточно высокую популярность среди пользователей. Как и положено любому свободному программному обеспечению, сборник «Open Software CD для Windows Spring Edition» распространяется за символическую плату, покрывающую стоимость носителя и пересылки, а также доступен для свободной загрузки с FTP-сервера проекта.

Новый вариант CD будет отличаться от своего предшественника. Помимо свободного ПО, распространяющегося исключительно по лицензии *GPL* или смежным лицензиям, в новый набор войдут и бесплатные программные продукты, в том числе выпускаемые коммерческими производителями, например Adobe Systems Incorporated. Подобный шаг объясняется желанием авторов проекта расширить функциональность входящего в сборник ПО. Как и первый вариант, новый сборник будет предназначен для распространения среди российских учреждений сферы образования, науки и культуры, бюджетных и некоммерческих организаций, не обладающих достаточными средствами для приобретения дорогостоящих коммерческих программ.

Linux-мобилизация

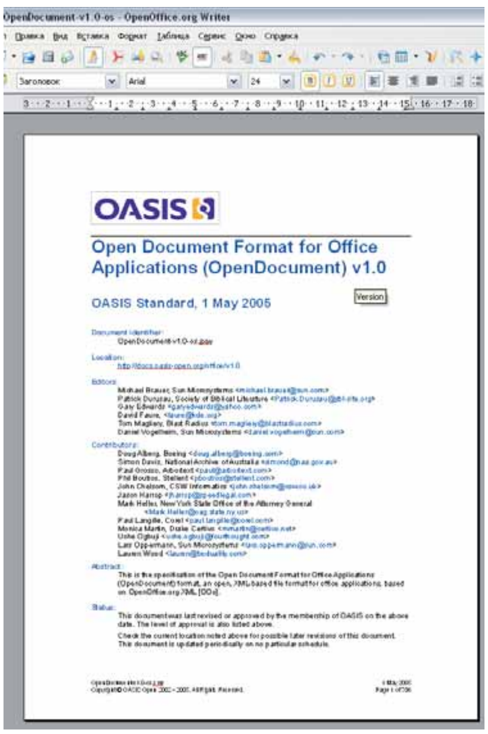
Defining standards for Linux based telephony

Образованный еще в 2005 году альянс *Linux Phone Standards* (LiPS) Forum, объединивший усилия таких известных компаний, как France Telecom, Telecom Italia, MontaVista Software, PalmSource (позднее переименованная в Access Inc) и многих других, опубликовал первые спецификации для Linux-разработчиков, нацеленных на мобильный рынок. В документе даются рекомендации по созданию интерфейса, адресной книги, функций голосового набора, использованию API ввода текста, а также некоторых сервисных функций и наборов виджетов. Стоит отметить, что, несмотря на активную работу в мобильной сфере норвежской компании Trolltech, участники данного альянса отдают большее предпочтение проекту GNOME, в связи с чем практически все рекомендации базируются на использовании средств разработки GTK. Объяснение данному факту очень простое: многие участники Форума используют GNOME в качестве базового ПО. К таким компаниям относятся, например, Purple Labs, Access и Open-Plug. Следующий выпуск спецификаций Форума будет нацелен на такие функции, как обмен мгновенными сообщениями и возможность изменения интерфейса пользователем или оператором сотовой связи. Третий набор спецификаций, выход которого намечен на конец текущего года, позволит разработчикам мобильных приложений более оптимально использовать различные аппаратные ресурсы телефонных трубок.

ODF: «штатная» ситуация

В последнее время тема перевода правительственных организаций на использование открытого стандарта OpenDocument стала одной из наиболее популярных. Многие европейские организации принимают соответствующие постановления и осуществляют переход на ODF; не меньшее внимание уделяется данному вопросу и в сенатах многих американских штатов. Между тем, американские народные избранники довольно осторожно подходят к принятию подобных законов, не желая принимать скоропалительных решений: как минимум в пяти штатах, включающих Калифорнию, Флориду, Техас, Орегон и Коннектикут, законопроекты о переходе на ODF были отклонены сенаторами. Причин тому несколько. Во-первых, инициативные группы, выдвигавшие подобные законопроекты, проявили явную поспешность, предложив сенаторам принять открытый формат в качестве стандарта для официального документооборота без каких-либо предварительных исследований. Естественно, осторожные ко всяким переменам парламентарии не рискнули слепо принимать на веру доводы инициаторов законопроектов. Вторая причина афишируется меньше, но не исключено, что именно она является более весомой: в приватных беседах некоторые сенаторы признались, что опасаются «санкций» компании Microsoft, активно препятствующей продвижению открытых стандартов в правительственные структуры. Причина подобного упорства тоже ясна: уже давно пакет офисных приложений *MS Office* является довольно ощутимым источником доходов редмондского гиганта, а переход государственных учреждений на открытый формат документов позволит данным многочисленным организациям внедрить альтернативные офисные пакеты, включая главного конкурента *MS Office* – свободный пакет офисных приложений *OpenOffice.org*.

Впрочем, руководитель основанного в Вашингтоне ODF Alliance Марино Маркич [Marino Marcich] не унывает и говорит, что битва за ODF только начинается. Несмотря на неудачи в ряде штатов, законопроекты о переходе на ODF продолжают выдвигаться на рассмотрение. В частности, демократы штата Нью-Йорк приступили к продвижению открытых стандартов в государственных учреждениях, но, глядя на примеры соседей, решили не торопиться, а начать с предложений провести исследование возможности и целесообразности перехода на ODF. Не исключено, что подобный поэтапный процесс продвижения открытых стандартов в штате Нью-Йорк окажется более успешным и подаст пример сенаторам других штатов.



LXF Репортаж

Open Source Forum@Interop Moscow 2007

30-31 мая в Москве, в комплексе «Форум Холл» состоялся Interop Moscow 2007 – Конгресс и Выставка корпоративных информационных технологий. **Иван Зорин** побывал на проходившем в его рамках втором Open Source Forum'е и готов поделиться своими впечатлениями с читателями *LXF*.

Форум был открыт докладом Брайана Белендорфа [Brian Behlendorf], одного из основателей проекта Apache, озаглавленным: «Ten Things you might not know about Open Source» [«Десять вещей, которых вы могли не знать про Open Source»]. В начале своего выступления Брайан напомнил присутствующим, что Open Source – это не только всем известные программы и приложения: существует свыше 100 тысяч разработок, каждая из которых занимает свою нишу.

Далее было отмечено, что открытое ПО предшествовало проприетарному, так как еще мейнфреймы продавались в комплекте с исходными кодами программ, расположенными на магнитных лентах; компания-покупатель могла беспрепятственно модифицировать уже установленное ПО под свои нужды. Потому идея свободного ПО, предложенная Фондом свободного программного обеспечения в 1985 году, была по своей сути не новой.

Затем г-н Белендорф напомнил несколько фактов об Apache. Проект был запущен в 1995 году и преследовал две цели. Первая, прагматическая – сконцентрировать усилия разработчиков; вторая, идеалистическая – сохранить HTTP как открытый стандарт. Основатели проекта были обеспокоены тем, что компании, владеющие как рынком серверов, так и рынком браузеров, начнут определять свои собственные технологии и спецификации. Но, несмотря на все эти опасения, вскоре после запуска Apache стал основным web-сервером и до сих пор удерживает пальму первенства.

В своем докладе Белендорф упомянул и другой известный открытый проект – BioPerl, который является свободной альтернативой исследованиям коммерческого консорциума Celera. Celera был основан как проект по расшифровке последовательностей генома человека. Все результаты исследований и все расшифрованные геномы патентуются консорциумом, чтобы впоследствии можно было контролировать доходы от применения этих знаний. Однако, независимо от этого, Джим Кент [Jim Kent], докторант из университета Санта Круз в Калифорнии, написал на языке Perl порядка десяти тысяч строк кода, которые представляют собой набор функций для работы с геномами и их последующей расшифровки. Получившийся проект и получил название BioPerl. Сегодня он может использоваться как самодостаточное приложение, а также включаться сторонними разработчиками в свои программы в качестве внешнего инструментария. Во время доклада была также упомянута распространяемая по лицензии Creative Commons книга Карла Фогеля [Karl Fogel] «Создавая программное обеспечение с открытым исходным кодом» [Producing Open Source Software], которая посвящена не столько технической, сколько социальной и психологической проблемам открытой разработки. Книгу можно найти по адресу: http://producingoss.com/.

«Open Source нужна ваша помощь». Этой теме была посвящена заключительная часть доклада Брайана Белендорфа. Чем вы можете помочь? Попробуйте использовать свободные приложения, такие как *Firefox*, *Thunderbird*, *OpenOffice.org*, не отказываясь от Windows или







Завершен конкурс на лучшую статью об OpenOffice.org

Подведены итоги конкурса на лучшую статью о пакете *OpenOffice.org*, проводимого при поддержке журнала *Linux Format* (см. **LXF89**). 1 июня 2007 года, в соответствии с положением о конкурсе компетентная комиссия определила победителей:

» в номинации «**Офисные технологии**» победила статья «*Сложная разметка и хитрости в OpenOffice.org Math*», автор Смирнов Д. В.

» в номинации «**Тесты, сравнения, обзоры**» победила статья «*Сравнение OpenOffice.Draw (StarDraw) с Visio*», автор Елисеев А. А.

» в номинации «**Учим работать с OpenOffice.org**» победила статья «*Лабораторные работы в OpenOffice.org*», автор Емельянов Э. В.

» в номинации «**История одного проекта**» победила статья «*Семь дней на пути к Ru.OpenOffice.org*», автор под псевдонимом Opener

» в номинации «**Об этом уже писали**» победила статья Сольвейг Хогленд (Solveig Haugland) «Использование стилей в *OpenOffice.org*», авторы перевода под псевдонимами John Lynx, George Yves.

Победителей ждут призы:

» Годовая подписка на журнал «*LinuxFormat*»;

» Премии в размере 10000 рублей, из призового фонда журнала «*Linux Format*» и компании «Инфра-Ресурс».

» Годовая подписка на журналы «Мир ПК», «Открытые системы» или еженедельник Computerworld по желанию победителя.

Более подробную информацию можно найти на официальном сайте конкурса: http://i-rs.ru/article/articleview/540/1/30/

От души поздравляем победителей!



Mac OS X – они доступны и для этих систем (*см. стр. 40*). Загрузите LiveCD-дистрибутив. Принимайте участие в дискуссиях на форумах, посвященным поддержке того или иного проекта – задавайте вопросы, если возникают трудности; помогайте остальным, если знаете ответы на вопросы, с которыми они сталкиваются. Находите ошибки и сообщайте о них разработчикам. Носители языка могут оказать неоценимую помощь по переводу и локализации программ, при этом совсем не обязательно обладать навыками программиста.

После доклада состоялась дискуссия с участниками Open Source Forum: Брайаном Белендорфом, Луи Суаре-Потом [Louis Suarez-Potts] из *OpenOffice.org* и Эриком Оллмэном [Eric Allman] из Sendmail, Inc., которые отвечали на вопросы зала. Ведущим этого мероприятия был Дмитрий Дмитриев (Линукс Инк), от которого для «затравки» и последовал первый вопрос: вынуждены ли открытые проекты вечно догонять более успешные проприетарные продукты?

Луи ответил, что не согласен с этим утверждением. По его мнению, открытые проекты всегда были источником инноваций и свежих идей. В качестве примера он привел возможность экспорта документов в формат PDF – функцию, которая была доступна с первых версий *OpenOffice.org* и которую Microsoft добавляет в свой офисный пакет только сейчас. Затронув вопрос о лицензиях на открытое ПО, Луи отметил, что *GPL3*, на его взгляд, имеет ряд ощутимых плюсов по сравнению со второй версией, так как охватывает патенты, что очень важно для мира Open Source.

«Иски против открытых проектов со стороны Microsoft – коммерческое самоубийство», – таким был ответ Брайна на вопрос по поводу обвинений Microsoft в адрес разработчиков ядра Linux и его пользователей в том, что открытый код нарушает патенты редмондского гиганта. И пояснил: «Тогда им придется показать свои исходные тексты, на что они не пойдут ни за что на свете».

Закончив отвечать на вопросы, «звезды Open Source» освободили трибуну для выполнения следующей части программы форума – дискуссии о плюсах и минусах открытого ПО. Состав ее участников был таким: Александр Давыдов (NAUMEN), Алексей Смирнов и Алексей Новодворский (ALT Linux), Владимир Мамыкин и Владислав Шершульский (Microsoft), Андрей Федоров (Digital Design); ведущим данного мероприятия был Дмитрий Шурупов (журнал «Системный Администратор», портал **www.nixp.ru**).

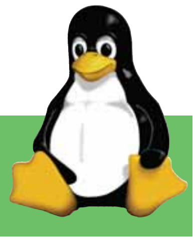
Владислав Шершульский отметил, что открытость хороша не везде, и есть такие проекты, в которых делегирование прав на доступ к исходному коду просто необходимо. Владимир Мамыкин назвал отчасти мифом и то, что открытые проекты более безопасны. В качестве примера он привел ядро Linux: с ростом популярности в нем якобы находятся все новые и новые «дыры»; на сегодняшний день это уже порядка трехсот уязвимостей. Также Владимир отметил, что Microsoft, в случае требований со стороны государства, может предоставить последнему к исходным текстам своих продуктов и операционных систем в доверительном окружении с целью их ревизии на предмет наличия каких-либо вредоносных «закладок».

Алексей Новодворский изящно парировал некоторые из этих утверждений. Когда приводят подобные цифры, сказал он, навряд ли они являются объективными и отражают реальную суть; с другой стороны, можно сравнить наличие уязвимостей, в том числе, критических, в двух web-браузерах – открытом *Firefox* и проприетарном *Internet Explorer*. По всем показателям, как объективным, так и субъективным, сравнение будет явно не в пользу последнего. «Насколько мы, граждане, можем доверять таким процедурам?»,– задал вопрос Алексей в ответ на упоминание Владиславом возможности проведения ревизий проприетарного кода. «Действительно ли это будет независимая экспертиза, а не экспертиза, проводимая фиксированной фирмой, назначенной государством для контроля?»

Владимир Мамыкин затронул также экономическую составляющую Open Source, в которой основную прибыль составляют дополнительные услуги и поддержка. Было отмечено, что, несмотря на динамичное развитие Open Source в нашей стране, действительно крупных открытых проектов национального масштаба как таковых нет. Алексей Новодворский же отметил отсутствие качественных отечественных специалистов в области операционных систем; так, ни на одной международной конференции за последние годы не было ни одного доклада по данной тематике, представленного российскими разработчиками. Как можно получить специалистов по операционным системам, обучая людей программированию, и при этом не имея доступа к исходным кодам приложений и самих ОС? В западных технических вузах на дисциплинах, связанных с компьютерными науками, обучение студентов проводится на открытых программах и операционных системах; и после обучения, обладая высокой квалификацией, многие из них становится лучшими разработчиками Microsoft. Было также отмечено, что открытые проекты бессмертны – а в случае с проприетарным ПО, если исчезает производитель, исчезает и его продукция.

Алексей Смирнов отметил роль важную патентов в мире Open Source. Когда патентное право только зарождалось, считает он, его целью было поддержать инновации и развитие изобретений, а сегодня для мира открытого ПО патенты являются своеобразным сдерживающим механизмом. В заключительной части своего слова Алексей отметил отсутствие методических пособий для школ, которые были бы применимы к открытым приложениям. Возможно, с их появлением внедрение открытых разработок в образовательный процесс пойдет быстрее.

Подводя итог мероприятию в целом, было приятно наблюдать, что в последнее время российскому Open Source-сообществу уделяют внимание мировые лидеры этого движения, которые осознают перспективы развития программного обеспечения с открытым исходным кодом у нас в стране. С другой стороны, судьба и проблемы OpenSource в России не безразличны и отечественным компаниям и деятелям, что также не может не радовать.



Новости короткой строкой

» Официально выпущена операционная система Fedora 7, впервые представленная в виде настольной и серверной версий, а также подготовленная в виде двух LiveCD, использующих в качестве графической среды KDE и GNOME.

» Фонд Mozilla выделил в качестве гранта на дальнейшее развитие $100 тыс. проекту свободного ТВ-плейера Democracy.

» Компания TurboLinux объявила о начале продаж управляемого Linux мини-компьютера Wizpy в США, Великобритании, Австралии, Индии и Сингапуре.

» Проект OpenSUSE приступил к разработке дополнительного CD “The openSUSE EDU-CD” (или “The SLEducator”), предназначенного для образовательных учреждений.

» Компания Innotek объявила о выходе новой версии *VirtualBox* (1.4.0), первой свободной профессиональной виртуальной машины, распространяющейся под двойной лицензией (*GPL*/закрытая).

» Проект *live.linuX-gamers.net* объявил о выходе основанной на *Slackware* тестовой версии *lg-live Live-DVD 0.9.2*, в состав которой вошло пятнадцать наиболее популярных трехмерных игр для среды Linux: *Enemy Territory, Torcs, True Combat: Elite, Warsow, World of Padman* и другие, при этом оригинальные драйверы от nVIDIA и ATI также представлены на диске и устанавливаются во время загрузки системы.

» Российская компания Etersoft объявила о выходе новой версии специализированных библиотек WINE@Etersoft 1.0.7, обладающей улучшенной поддержкой отечественных Windows-приложений.

» Вышла очередная версия свободного пакета офисных приложений *OpenOffice.org 2.2.1*, а почти одновременно российская компания Инфра-ресурс подготовила собственную сборку *OpenOffice.org 2.2.1* Pro, адаптированную к работе в российских условиях.

» Компания Microsoft объявила о повышении цен на свои программные продукты, распространяемые в России, в среднем на 25%.

LXF Репортаж

Мультиплатформенный школьник XXI века?

На каком программном обеспечении будут учиться следующие поколения россиян? Что ждет российскую школу в связи с усилением борьбы с пиратством? Что такое «свободное ПО»? Эти вопросы обсуждали в «Росбалте» учителя, чиновники и разработчики.



Ася Власова, ИА «Росбалт»

4 июня 2007 г. в пресс-центре информационного агентства «Росбалт» прошел круглый стол «Городское педагогическое собрание – за свободное ПО в школах». Мероприятие было организовано ИА «Росбалт» и недавно созданной IT-структурой – компанией «iTREND».

В дискуссии приняли участие представители городской администрации, федеральной законодательной власти, педагоги, разработчики ПО, эксперты.

В последнее время все чаще ведутся разговоры о необходимости создания и внедрения в России современной отечественной операционной системы (ОС). Перспективами российской ОС заинтересовались уже не только сами разработчики, но и госструктуры, законодатели, представители бизнеса, школы и вузы. Для последних создание доступной по цене и при этом абсолютно легальной системы будет означать решение проблемы обучения современным информационным технологиям. Пока создание российской ОС является реальной, но отдаленной перспективой, школам приходится решать эту проблему самостоятельно.

Активизация процесса подключения школьных компьютеров к Интернету, отсутствие отечественной свободной оперативной системы, недостаточность законодательного регулирования в этой сфере, – все это ставит перед властями много вопросов, подчеркнула в ходе работы круглого стола депутат Госдумы Валентина Иванова. По ее мнению, случай с директором Сепычской средней школы Верещагинского района Пермского края Александром Поносовым, который был привлечен к суду за использование нелицензионного программного обеспечения (ПО), говорит о том, что времени на решение этих вопросов немного.

Иванова отметила, что Всероссийское педагогическое собрание, сопредседателем-координатором которого она является, выступило в защиту Поносова, обратившись в Генпрокуратуру. Педагог уже полтора года доказывает свою невиновность в судах, и в подобной ситуации может оказаться не только любой учитель, но и люди других профессий, имеющие дело с программным обеспечением, отметила Иванова.

Как правило, в российских школах «по умолчанию» используют программные продукты Microsoft. В большинстве случаев они являются «пиратскими», также школьники не получают разностороннего освоения IT, владения различными ОС: в частности, многие преподаватели информатики в российских школах не имеют адекватного представления о возможностях открытых продуктов. Вместе с тем уже сейчас существуют программные решения на основе Open Source, которые могут быть применены в учебно-образовательных программах. Сам Александр Поносов, являвшийся, по сути, «гвоздем» круглого стола в «Росбалте», отметил, что хотя в условиях консервативной школьной среды переход на альтернативные операционные системы затрудняется, однако при соответствующем желании особых сложностей в освоении, например, Linux, возникать не должно. У него самого при чрезвычайно высокой занятости освоение новой системы заняло не больше месяца.

По мнению аналитиков, в России достаточно низкий уровень осведомленности о свободном ПО, а доступность дешевого пиратского «софта» не оставляет никаких препятствий к тому, чтобы пользователи устанавливали на свои ПК продукты Microsoft. Что касается школ, то анализ Мининформсвязи показал: до 75% персональных компьютеров работают на нелицензионном общесистемном и прикладном программном обеспечении. При этом общее число используемых в настоящее время персональных компьютеров составляет порядка 675 тыс. штук. Такой уровень пиратства, к тому же в системе образования, т.е. в государственных учреждениях, сегодня становится существенным препятствием для вступления России в ВТО, одним из условий которого является соблюдение прав на интеллектуальную собственность.

Напомним, что недавно Мининформсвязи выступило с инициативой, направленной на легализацию ПО, используемого в школах. Ведомством был сформирован базовый перечень общесистемного и прикладного ПО для закупки и установки на школьные персональ-





ные компьютеры. Кроме Microsoft, в него вошли программные продукты «Лаборатории Касперского», «Доктор Веб», Adobe и Corel. Стоимость такого пакета для одного ПК составил бы 787 руб. в год. В мае этого года Министерство финансов отказалось выделить средства из федерального бюджета на реализацию этой программы. Специалисты уверены, что при закупке прикладных программ для школ важно требовать многоплатформенности, то есть возможности использования ПО хотя бы под двумя ОС – Linux и Windows, – чтобы избежать зависимости от монопольного поставщика.

О том, что после проведения специальных информационно-обучающих семинаров ряд петербургский школ выразили заинтересованность в переходе на Linux, сообщил ведущий специалист комитета по образованию правительства Санкт-Петербурга Игорь Базлов.

По его словам, осенью на базе этих школ будет создана экспериментальная площадка, где будет внедряться Linux. Если опыт даст положительные результаты, комитет будет рекомендовать Linux для внедрения во всех школах, отметил Базлов.

Все участники дискуссии были едины во мнении о том, что учащиеся и учителя должны иметь альтернативные способы изучения IT-технологий, овладевать знаниями о разных программных продуктах и операционных системах, что практически не реализовано на данный момент. Одной из главных проблем собравшиеся отметили низкую осведомленность самих преподавателей. Александр Поносов отметил, что он сам до января 2007 г., когда против него был возбуждено уголовное дело об использовании нелегального ПО в школе, не знал о существовании Linux – наиболее популярной свободной ОС.

В настоящее время «дело Поносова» пока еще не закрыто (14 июня должно состоятся очередное заседание суда) и даже имеет любопытное развитие: в конце мая автономная некоммерческая организация общественный центр «Равенство» (Пермский край) направила в Генпрокуратуру России заявление с требованием привлечь к уголовной ответственности прокурора Верещагинского района Александра Троянова. Как сообщил «Интерфакс», работники прокуратуры Верещагинского района в ходе расследования уголовного дела в отношении директора Сепычевской средней школы Александра Поносова, обвиняемого в использовании контрафактных программ Microsoft, сами использовали нелицензионное программное обеспечение. Сам Александр Поносов отметил, что планирует «сражаться до последнего», а активную поддержку надеется получить от общественности, разработчиков и депутатов ГД РФ, которые, впрочем, уже немало помогли учителю. Валентина Иванова подтвердила готовность и дальше отстаивать права Поносова в непростом и прецедентном для России деле.

По мнению Галины Пожариной, изучение открытого ПО в школах будет воспринято учениками с большим интересом, как все новое, а также будет способствовать использованию свободного легального софта и в домашних условиях, что крайне важно для разностороннего образования школьников. Даже в том случае, если школа имеет лицензионное ПО от Microsoft, необходимо устанавливать на школьных компьютерах Linux как вторую операционную систему и давать ученикам полное представление о современном мире ПО.

Павел Житнюк отметил наличие большого количества свободных программных продуктов образовательного характера, которые могут быть использованы на уроках алгебры, геометрии, астрономии, физики и т.д., а также программы для начальной школы. Сейчас только порядка 5% госучреждений в России использует легальное ПО, а повсеместное распространение коммерческих продуктов нескольких или даже одной компании наносит ущерб как отечественной IT-отрасли, так и информационной безопасности всего государства. Существующие «мифы» о трудностях использования Linux уже давно развеяны: на данный момент эта система представляет полноценную альтернативу продуктам Microsoft (простая установка, удобный графический интерфейс, удобные офисные приложения, огромное количество прикладных программ, более низкая цена лицензии и техподдержки, отсутствие компьютерных вирусов, открытость кода и возможность гибкой модификации под задачи пользователя, совместимость с Windows-файлами, и пр.).

Леонид Сомс привел такие цифры: экономия для одной школы при использовании открытых продуктов может составить около 800 тысяч рублей. К началу учебного года компания «Линукс-Инк» планирует сделать подарок всем петербургским школам: предоставить лицензионный дистрибутив, адаптированный компанией – Scientific Linux.

Однако этот же эксперт отметил, что ситуация не столь уж безоблачна. Процесс внедрения в организациях, в частности, в школах, свободного программного обеспечения длителен и сопряжен с рядом проблем. Одна из важнейших – инертность преподавательского сообщества, выраженная в затрудненном переходе от привычных образовательных программ к новым. Другая проблема заключается в низкой совместимости Linux и Windows в плане администрирования. «Обучать на Linux, а администрировать на Windows – это нонсенс», – полагает Сомс.

В этом с ним согласен директор ГОУ школа N369 Красносельского района Санкт-Петербурга Константин Тхостов. Он отметил: «Если пользователь, как физическое лицо, сам решает, какой операционной системой пользоваться, то юридическое лицо, такое как школа, работает в системе Горзаказа, с различными комитетами администрации, в частности, Комитетом финансов. С переходом на Linux эта работа будет затруднена». Вместе с тем Тхостов полагает, что переход к 1 сентября на Linux был бы лучшим подарком школьникам ко Дню знаний.

Игорь Базлов, в свою очередь, отметил, что Петербург в области IT всегда немного опережает остальные регионы России. Сейчас почти по всех учреждениях города так или иначе используются свободные программные продукты и OpenOffice.org. Однако для дальнейшего развития в этом направлении требуется поддержка из Москвы.

LINUX FORMAT Обзоры



Новинки программного и аппаратного обеспечения в описании наших экспертов



Алексей Федорчук
Свою первую (и последнюю) программу написал еще на Алголе

Linux — это серьезно?

Сполдюжины лет назад один редактор одного компьютерного издания, отметив, что линуксоиды только и делают, что ругают Microsoft, добавил: всерьез можно будет относиться к Linux'у только тогда, когда, наоборот, Microsoft начнет его ругать.

Этот знаменательный момент наступил. Сначала появились аналитические обзоры сравнительной оценки совокупной стоимости владения Windows- и Linux-платформами, потом их результаты (разумеется, благоприятные для Microsoft – ведь кто девушку ужинает, тот ее и танцует, не так ли?) перекочевали на рекламные полосы компьютерных (и даже не очень компьютерных) журналов. И, наконец, апофеоз маркетингового гения самой великой маркетинговой компании: рекламная акция под названием «продам Linux-сервер».

Не знаю, на кого это рассчитывали в мировом масштабе, но на Руси реакция на нее была однозначно юмористическая. Российское представительство Microsoft завалили предложениями этот самый Linux-сервер купить. А его сотрудникам, бедолагам, пришлось отвечать на эти запросы на полном серьезе, в результате чего реклама эта постепенно исчезла. А может быть, просто раскупили весь наличный запас Linux-серверов?

Так что настало время относиться к Linux вполне серьезно.

alv@posix.ru

Сегодня мы рассматриваем...

EnergyXT с. 16



› EnergyXT предлагает эластичное растяжение времени и другие запредельные музыкальные эффекты.

Cedega с. 17



› – Новая коллекция Кейт Мосс весьма неплоха, не так ли?
— Как ты сказал: Кейт кто?

НАШ ВЕРДИКТ: пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибальной шкале (10 – высшая оценка, 0 – низшая). Как правило, мы оцениваем функциональность, производительность, простоту использования и цену, а для бесплатных программ учитывается документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.

Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но, если разработчики рекомендуют Autopackage, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth
Разработчик: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Документация	9/10

» *Если весь мир – сцена, то Google Earth – театр. Простая в использовании, захватывающая и ободряюще практичная программа.*

Рейтинг 9/10

Dofus

Захватив боевой топор, **Энди Хадсон** отправляется искать приключений в новом мире. Хотите последовать за ним?

Вкратце...

» Ролевая онлайн-игра для множества игроков на основе Flash. Реальных альтернатив нет, но среди других Linux-совместимых RPG Second Life находится ближе

Ах, эти юношеские дни, проведенные за ролевыми играми! Если в восьмидесятых вам было 10–15 лет, вы, наверное, помните *HeroQuest*, фантастическую пошаговую ролевую игру (RPG), захватившую тогда всю Британию. В ней вы могли стать одним из четырех персонажей, предпринимавших походы для поиска сокровищ или уничтожения неких злых сил. Что было здорово в той игре – миссия средней продолжительности занимала около часа, поэтому за обеденный перерыв можно было еще и поесть.

Спустя 20 лет ролевые игры существенно углубились в виртуальный мир. Гиганты уровня *World of Warcraft* и *Star Wars: Galaxies* стали местом, где игроманы всего мира реализуют свои ролевые фантазии. В этих мирах полно миссий, от тусовки с инопланетянами в забегаловке до линчевания гоблинов. Успех общеизвестной *Second Life* окончательно убедил несколько очень крупных компаний в необходимости присутствия в виртуальном пространстве.

Однако во все упомянутые игры не так-то просто «сыграть по-быстрому», когда у вас мало времени. Зато в *Dofus* легко можно нырять и выныривать обратно, когда захочется. *Dofus* – это RPG французской разработки, весьма популярная среди наших галльских



› Приятно встретить в Dofus древнейшие профессии. Перед вами мастерица по изготовлению луков, предлагающая свои услуги за деньги. В остальном языковые навыки игры плоховаты.

кузенов. Выпущенная в сентябре 2005 г., на сегодняшний день игра поддерживается почти трехмиллионным сообществом любителей во всем мире. Британский вариант – это возможность для Ankama, компании-разработчика Dofus, протестировать некоторые новые функции перед потенциальной конвертацией игры в другие локали [русской среди них пока нет, – прим. ред.].

А что же сама игра? *Dofus* предпочла псевдореалистичную графику *World of Warcraft* и *Star Wars: Galaxies* тотальной Flash-анимации. Ankama позиционирует *Dofus* как почти интерактивный мультфильм. Использование Flash означает, что качество графики не зависит от размеров окна, и это определенно придает игре гладкую мультяшную приятность. Читатели, знакомые с историями про Астерикса и Обеликса, найдут *Dofus* поразительно знакомым с графической точки зрения. Здесь даже есть персонаж по имени Догматикс!

Статистикс

Сюжет простой: вы ищете приключений в провинции Amakna. Ваша цель – найти шесть Dofus'ов, или Драконовых яиц, они наделят вас неограниченной властью. Но способ добиться цели зависит только от вас: можно входить и выходить из игры в пределах пятиминутных кусочков, а можно слоняться там часами, исследуя каждую пядь предоставленного виртуального мира. Удобно, что *Dofus* предлагает новичкам пробный режим под названием Incarnum, в котором можно потренироваться перед полным погружением в игру.

Исследуем интерфейс Dofus

Неиграющие персонажи
Персонажи с восклицательным знаком над головой наставят вас на путь истинный.

Аватары
Персонаж, представляющий вас, вы выбираете сами. Все, что персонаж говорит, всплывает в пузыре над его головой.

Значки
Доступ к вашему арсеналу, дневнику миссии, картам и заклинаниям.

Компас
При поиске цели можно обратиться к удобному маленькому компасу. Цифры в сердце указывают уровень здоровья.



Dofus не похож на большинство современных игр тем, что здесь нет свободного развития сюжета. Поле игры состоит из множества экранов, наводящих на мысль о графическом эквиваленте старых текстовых игр (North, West и т.п.).

Чтобы указать выходы, существующие на данном экране, *Dofus* использует маленькие значки, напоминающие компас: щелкните на любом из них, и ваш персонаж перейдет с данного экрана на другой, в зависимости от направления стрелки компаса. К несчастью, эти значки трудновато разглядеть из-за неудачного выбора цветов, сливающегося с фоновым рисунком. Чтобы найти подобную штучку, иногда бывает необходимо потратить несколько минут. Понятно, что создатели игры не хотели делать эти значки броскими, но ясность все же должна иметь преимущество перед желанием соблюдать цветовую схему.

Взаимодействовать с другими игроками через чат очень удобно. Если вы находитесь в Incarnum, можно воспользоваться функцией Incarnum и разослать свое сообщение всем игрокам, глобально. Однако по умолчанию чат ограничен персонажами, находящимися в пределах видимости, и вместо текста в окне чата высказывания персонажей всплывают как пузыри у них над головами. Это неудобно, когда несколько человек пытаются разговаривать одновременно, так как теряется последовательность разговора; приходится все равно обращаться к чату, и пузыри остаются не у дел.

В игре присутствует много неиграющих персонажей (NPC), которым можно задавать вопросы и получать ответы на них. Если над головой одного из таких персонажей – зеленый восклицательный знак, то это обычно означает, что он хочет вам что-то сказать или, чаще, подсказать какое-либо действие. Внутри Incarnum это может быть предложением набрать воды для нимф, попросить материю, из которой кто-нибудь сделает флаг, и даже умереть, чтобы посмотреть на жизнь с другой стороны! Это делается, конечно, для того, чтобы облегчить игру, и нужно провести немало времени в Incarnum, прежде чем осмелиться войти в город Astrub.



› Вдали появился Дарт Вейдер (угрожающая черная фигура в кружке), правда, на сей раз без своего лазерного меча. Пора одеться и вооружиться!

Dofus предоставляет социальный опыт, но можно безоглядно вступить в игру и положиться на удачу. Помните, что для истребления самых опасных врагов необходимо объединять усилия с другими игроками. Самое страшное, что может встретиться вам в Incarnum – это Gobball War Chief (Главный военный вождь), *Dofus*-версия волка в овечьей шкуре. Не позволяйте их пушистой наружности усыпить вашу бдительность: они больно кусаются!

Поправочки-с

Но есть некоторые вещи, умаляющие достоинства *Dofus*. Самое очевидное – тексты, используемые в игре. Хотя игра объявлена специально предназначенной для британского рынка, тексты страдают ошибками перевода – очевидными кальками с французского. Если вы когда-нибудь пользовались приложениями, переведенными с других языков, то вам, без сомнения, знаком тот квази-английский, который получается в результате. Ankama открыта для игроков, присылающих поправки – а может быть, им стоит обратить внимание на Launchpad от Canonical, чтобы быстро найти подходящий перевод.

Другой недостаток – отсутствие у графики всякой глубины: вместо того, чтобы подниматься и спускаться по лестнице, персонаж просто перемещается по сетке, что может приводить к нереальным ситуациям, когда он просто зависает над землей.

Но все это мелочи. Игровое пространство обширно, а разработчики обещают нам все новые приключения в будущем. Самое главное, доступ к *Dofus* можно получить бесплатно, хотя количество серверов для этого ограничено. Но даже стоимость годовой подписки для полного доступа в €60 большинству игроков не покажется чрезмерной.

Итак, можем мы рекомендовать *Dofus* или нет? Зависит от ваших ожиданий. Если вам больше нравится полное погружение, то оставайтесь лучше с *Second Life* или *World of Warcraft*. Но новичкам в ролевых играх, несомненно, стоит попробовать *Dofus*. Независимо от уровня игровой квалификации, вы вступите в живое игровое сообщество и получите захватывающий сюжет, с которыми в ближайшие несколько месяцев уж точно не соскучитесь. LXF

Системные требования

Хотя игра и основана на Flash, но и у нее есть минимальные и рекомендуемые системные требования. Непонятно, почему обделили пользователей PowerPC Mac, указав лишь требования к PC. Dofus без запинки работал на нашем компьютере с 2.8-ГГц Pentium 4, гигабайтом ОЗУ и широкополосным подключением, используя Ubuntu 7.04 и Flash-модуль, добытый в репозитории Multiverse.

Минимальные требования:
- Pentium III 700МГц
- 128МБ ОЗУ
- 56K модем

Рекомендуется:
- Pentium III 1ГГц
- 256МБ ОЗУ
- Широкополосное подключение

LINUX FORMAT Вердикт

Dofus

Разработчик: Ankama
Сайт: www.dofus.com
Цена: €60 за годовую подписку, возможны варианты.

Сюжет	8/10
Графика	6/10
Продолжительность	10/10
Цена	10/10

› *Приятная игра, нам она очень нравится – но есть еще, над чем поработать.*

Рейтинг 8/10

HP LaserJet 2700n

Вы просили побольше об устройствах? Получайте: **Пол Хадсон** исследует новейший цветной лазерный принтер бюджетного класса…

Вкратце...

» Цветной лазерный принтер, не разрушающий семейный бюджет до покупки картриджей. Можно сравнить с любым хорошим струйным принтером.

Интересно, одинаково ли один и тот же принтер работает под Linux и под Windows? По уверениям HP, ее драйверы принтеров содержат код, раскрыть который компания не может, поэтому разумно было бы предположить, что в Linux принтер будет работать хуже, чем на Windows или OS X. А вот и нет: LaserJet 2700n, новейший цветной лазерный принтер HP бюджетного класса, печатает почти идентично на любой предложенной платформе. Более того, настроить его под Linux гораздо проще.

Как все современные принтеры, LaserJet 2700n «понимает» Ethernet и USB. Это уже звучит обнадеживающе для Linux-пользователей: ведь, как и многие другие устройства, принтеры, подключенные к Ethernet, «просто работают», не требуя для себя особых драйверов.

Дело упрощается еще и тем, что для HP есть особая панель инструментов, предустановленная в Ubuntu, которая просканировала нашу сеть на предмет наличия принтеров, нашла LaserJet и автоматически организовала его взаимодействие со всеми нашими программами (Fedora тоже сработала без запинки).

Панель автоматически считывает уровни тонеров в каждом картридже и напоминает номер модели, чтобы при снижении уровня можно было заранее заказать замену. Да, на Windows все эти инструменты тоже есть, но их установка занимает около 10 минут и требует копирования на винчестер около 100 МБ данных. За это время мы распечатали несколько десятков пробных страниц на Linux!

Еще одна тонкая штучка – доступ к системе управления принтером через встроенный web-сервер, позволяющий настроить качество печати, параметры бумаги и параметры сети. Так как экран на устройстве слишком мелкий и читать на нем трудно, web-интерфейс – более удобный способ навигации в массе доступных параметров.





Майк считает...

«За $600 хотелось бы чего-то покруче LCD-дисплея как у калькулятора и четырех крошечных навигационных кнопок».

Чистая победа

Нужна скорость? 20 черно-белых и 15 цветных страниц в минуту вас вполне устроят. Оборотная сторона такой скорострельности – неимоверная быстрота, с которой расходуется тонер: расходы на замену картриджей придется подсчитывать чуть ли не по окончании настройки. И здесь вас ожидает шок: каждый из четырех картриджей обойдется примерно в $150. Да, да, полная замена будет стоить столько же, сколько сам принтер.

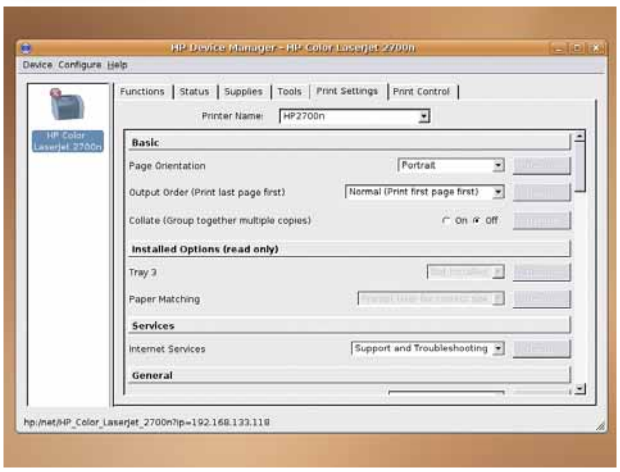
Будем справедливы, те же накладные расходы понесут и пользователи Windows. Но вернемся к самому интересному: будет ли отличаться качество печати на Linux по сравнению с Windows и OS X? Мы распечатали несколько пробных страниц с текстом и фото и раздали их всем окружающим на «дегустацию».

И что в итоге? Большинство не смогло заметить разницу. Печать обычного текста под Linux столь же безупречна, как в других ОС. Мы заметили некоторое ухудшение качества при печати изображений. Но даже это спорно, так как причиной могли быть недостатки ПО: *F-Spot* и *Gimp* хороши, но… все-таки не *Photoshop*. Кроме того, различия были настолько малозаметны, что никто их не обращал на них внимания до тех пор, пока мы сами не ткнули пальцем в легкое искажение мелкой детали одной из тестовых страниц.

HP достигла такого качества, поместив добрую часть кода своего драйвера в ядро Linux, а также вложив прогрессивные технологии цветной печати в само устройство, отчего выиграли все. Именно для этого в принтер встроен процессор 300 МГц, а при большой нагрузке на печать мы рекомендовали бы установить 64 МБ ОЗУ. Та часть драйвера, которую HP не смогла открыть, видимо, не оказала существенного влияния на результат.

LaserJet 2700n можно приобрести за $560 без НДС [в РФ – в районе $700, – прим.ред.], и это безумно мало, с учетом скорости и качества работы устройства. Картриджи дороги, но зато получается превосходный текст и высококачественные изображения (пусть не струйного качества), а настроить принтер может даже ребенок. LXF

› **Пробные страницы на Linux вышли не хуже, чем под Windows и OS X. Были бы только картриджи подешевле...**



› **Инструментальная панель HP разыскала устройство и автоматически настроила его. Для более тонкой настройки пользуйтесь web-интерфейсом.**

LINUX FORMAT Вердикт

HP LaserJet 2700n

Производитель: HP
Сайт: www.hp.com
Цена: $700 без НДС

Функциональность	8/10
Производительность	9/10
Простота использования	9/10
Цена	7/10

» *Безупречная машина. При умеренной цене на картриджи могла бы выйти в мировые лидеры.*

Рейтинг 8/10

Mandriva 2007 Spring Free

LINUX FORMAT На DVD

Metisse способен вскружить голову, но что стоит за внешним блеском? Интересуется **Ник Вейч**.

Вкратце...

» Передовой дистрибутив для рабочего стола – см. подробности на стр. 32!

Некогда, желая получить новейшее ПО, да такое, чтоб работало (чаще, чем нет), все обращались к Mandrake. Пока другие дистрибутивы дожидались стабильности ПО и всесторонней проверки обновленных ядер, команда Mandrake много работала над включением в дистрибутив последних новинок, твердо веря в то, что их можно заставить функционировать. Теперь это лишь воспоминания. Такие дистрибутивы, как Fedora и Ubuntu, обрели известность благодаря удобству и грамотному подбору ПО, в то время как версии Mandriva постепенно утратили былой задор и остроту. Сможет ли 2007 Spring Free исправить положение?

К сожалению, нет, хотя без новинок не обошлось и на этот раз.

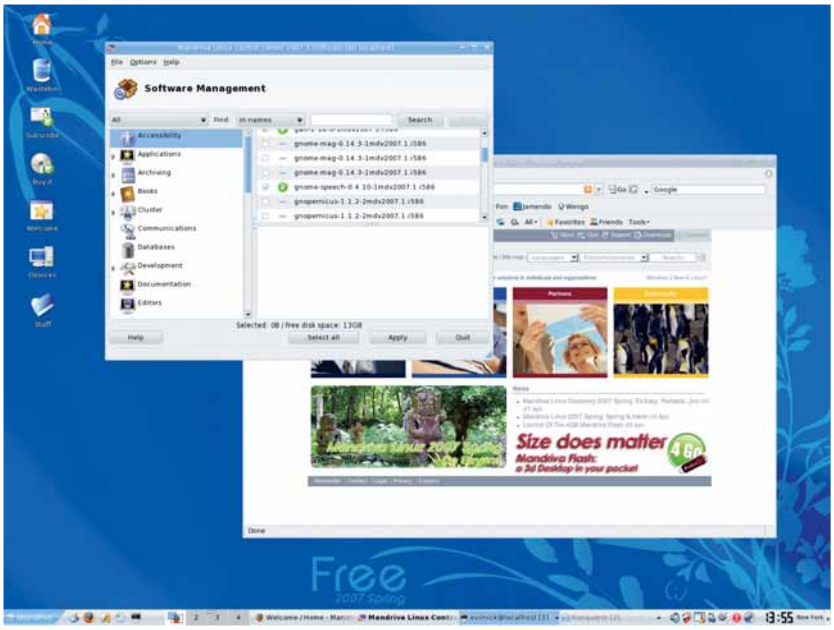
Прыжок вперед

Гвоздь программы – *Metisse*, собственное изобретение Mandriva для трехмерных рабочих столов. Надо сказать, *Metisse* – это не просто украшение, а попытка усовершенствовать продуктивность работы. Здесь применяется несколько новых функций, включая девять виртуальных рабочих столов, способных действовать, как один большой экран. Но вот в пользе вращающихся окон и прочих прибамбасов мы не совсем уверены.

Наиболее заметная проблема версии Free – то, что широко разрекламированные преимущества Mandriva в действительности не так легко получить. Возьмем *Metisse*. Сведения о нем появляются уже на этапе инсталляции.

> «Приятная новинка – Drakvirt, менеджер ПО виртуализации.»

Есть даже специальный сеанс настройки *X* для работы с ним (или для доводки *Beryl/Compiz*). А потом оказывается, что работать с *Metisse* может не всякая видеокарта, и похоже, что для свободной версии выбор ограничивается чипсетами Intel. Предельно ясно, почему не включены драйверы Nvidia (иначе версия не была бы свободной), но выяснять, «почему оно не работает», предоставляется самому пользователю. Вариант Powerpack, в котором драйверы Nvidia присутствуют, должен работать без проблем, для бесплатной же версии необходимые драйверы придется, как всегда, загружать самому.



› Все гладко и удобно, но нет прежнего напора. Неужели так – лучше?

Приятная новинка – *Drakvirt*, менеджер ПО виртуализации. В отличие от соперников, он умеет настраивать разные решения, включая *VirtualBox* и *VMware*, хотя для серьезной работы *Xen* от Red Hat все же его превосходит.

Остальное – не столько инновация, сколько коммерциализация. Интеграция ПО для телефонных переговоров Wengo – это хорошо, но только для тех, кто согласен оплачивать счет Wengo.

Для любого дистрибутива жизненно важны повседневные задачи: можно ли правильно настроить свое оборудование, инсталлировать ПО и добиться реальной продуктивности? Mandriva использует *Urpmi* для управления пакетами. Это прекрасный инструмент для выбора ПО в процессе инсталляции, но он несовершенен. Существуют проблемы с установкой в качестве обязательных даже таких зависимостей, которые всего лишь желательны. Каталог *Urpmi* стремительно устаревает, как и актуальные репозитории, что приводит к проблемам при загрузке нового ПО – это особенно заметно в первые несколько недель после выхода версии.

Отчаянный смельчак легко может загрузить пакеты Cooker для Mandriva, где всегда есть новейшие версии ПО. По крайней мере, были. Падающая популярность Mandriva приводит к тому, что количество новых пакетов постоянно сокращается.

Но Mandriva еще повоюет. Очевидно, что умы, стоящие за ним, умеют делать хорошие дистрибутивы. Чтобы получить превосходный, нужно только чуть поднажать. LXF

LINUX FORMAT Вердикт

Mandriva 2007 Spring Free

Разработчик: Mandriva
Сайт: www.mandriva.com
Цена: Бесплатно под GPL

Функциональность	8/10
Производительность	8/10
Простота использования	8/10
Документация	8/10

» *Крепкий дистрибутив с несколькими реальными новшествами, но сделать предстоит еще очень многое.*

Рейтинг 8/10

EnergyXT 2

Освободите музыку из плена треков и фейдеров. **Грэм Моррисон** рассматривает программу, создающую мелодию прямо на экране.

Вкратце...

» Модульный музыкальный комплекс с эффектами реального времени и программными синтезаторами. См. также: *Ardour* или *LMMS*.

Сначала *EnergyXT* была первоклассным инструментом для MS Windows. С ее помощью можно было собрать коллекцию программных синтезаторов и аудиоэффектов в единственном окне, используя вывод одного процесса в качестве входа для другого, как это делается в модульном синтезаторе. Полученные творения можно было затем использовать как источники звука для программ звукозаписи или даже самой *EnergyXT*, свободу творчества ограничивала лишь производительность процессора. *EnergyXT* принесла своему автору, Йоргену Аазе [Jorgen Aase], большой успех, а он уже планировал необычное продолжение: поддержку Linux.

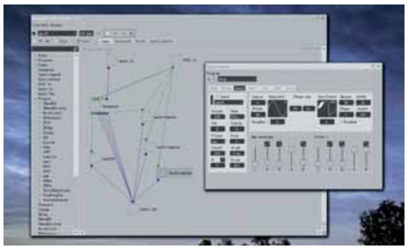
И вот перед нами специальная Linux-версия *EnergyXT 2* – правда, она выглядит

> «Открывает целый мир возможностей, и вы от него никогда не откажетесь.»

несколько пугающе. Основное окно выглядит примерно так же, как окно аранжировки в большинстве подобных программ, от свободной *Ardour* до индустриального лидера *Ableton Live*. Дорожки аудио, ударных и MIDI-данных располагаются друг над другом, а данные дорожек блоками уходят вправо. А вот и то, что приближает *EnergyXT 2* скорее к *Ableton Live*, чем к *Ardour*: данные каждой дорожки можно редактировать прямо из окна аранжировки, не надо открывать всякий раз новое окно. Это делает работу интуитивно понятной, а интерфейс удобным. Например, во время создания дорожки ударных можно добавлять каждый удар одним щелчком, а громкость регулировать, поднимая и опуская ползунок. На дорожке MIDI можно визуально отображать ноты и данные контроллера в одном окне, можно даже нажимать на клавиши виртуальной клавиатуры, чтобы воспроизводить ноты в реальном времени. При необходимости более тонкого контроля, двойной щелчок на области данных откроет окно точного редактирования расположения и продолжительности нот.

Творческое конструирование

Главное достоинство *EnergyXT* – унаследованный от предшественницы модульный принцип строения. Переключившись с окна аранжировки (под названием Sequencer) на окно графиков (Comps), вы увидите графическое представление своей настройки. Каждая дорожка – модуль со входом и выводами для аудио и MIDI-данных. Виртуальные кабели соединяют модули друг с другом и с секвенсором, но эту раскладку нетрудно изменить по своему вкусу. В *EnergyXT* встроено три модуля: секвенсор, используемый в окне аранжировки, модуль эффектов и синтезатор/сэмплер. Синтезатор хорош, его можно редактировать в реальном времени, меняя тембр звучания от резких отрывистых звуков до теплых, похожих на аналоговые, всплесков. С помощью модуля эффектов звуки обогащаются прерываниями, реверберацией, фазовыми и хоровыми эффектами. Кому этого мало, может загрузить множество других эффектов и синтезаторных модулей, полученных перекомпиляцией под Linux модулей Windows VST. Особенно привлекательна коллекция 'MDA' Пола Келлета [Paul Kellet], содержащая широкий диапазон аудиоэффектов и пару замечательных синтезаторов.



› Конфигурацию виртуальных подключений всей оснастки можно свободно менять.

Существует обратная зависимость между гибкостью программы и простотой ее использования, поэтому работать с приложениями такой модульности всегда сложно. Просто стянув все модули в окно аранжировки, вы обманете себя мыслью о том, что это всего лишь обыкновенная музыкальная программа. Но одно лишь нажатие кнопки Comps откроет вам целый мир возможностей, от которого вы никогда не откажетесь ради обычного ввода ноты за нотой. Может быть, вы и не закончите свою симфонию, зато получите массу удовольствия в процессе творчества. LXF

Изучаем интерфейс EnergyXT

Редактор ударных Удары добавляются простыми щелчками на дорожке ударных.

Сэмплер Каждый удар барабана имеет собственный сэмпл.

Запас эффектов Звуки можно украсить различными аудиоэффектами.

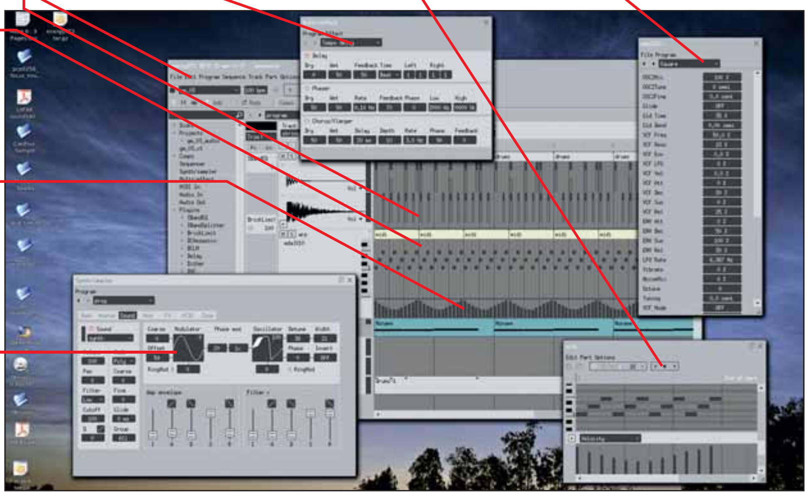
Матричное окно Чтобы точнее управлять редактированием, откройте новое окно.

Синтезатор JX10... ...нет, все-таки загрузка инструментов по выбору – это здорово.

Матричный редактор Ноты можно добавлять и редактировать в главном окне.

Управление данными Можно редактировать кривые громкости и другие управляющие данные.

Синтезатор реального времени В EnergyXT встроен свой программируемый синтезатор.



Linux Format Вердикт

EnergyXT 2

Разработчик: XT Software
Сайт: www.energy-xt.com
Цена: €39

Функциональность	9/10
Производительность	9/10
Простота использования	7/10
Документация	9/10

» *Использует эффекты реального времени и синтезаторы в той же манере, какой годами наслаждались пользователи Windows и OS X.*

Рейтинг 9/10

Cedega 6

Последние двенадцать месяцев **Грэм Моррисон** страдал по *Oblivion*. Благодаря новой версии *Cedega* он, кажется, получит новую игрушку.

Вкратце...

» Играйте в Windows-игры под Linux. В данной версии добавлена поддержка *Battlefield 2142*, *World of Warcraft* и *Madden 07*. См. также: *Crossover Linux* от *CodeWeaver*.

Восемнадцать месяцев назад мы рассматривали 5-ю версию *Cedega*. Благородный труд TransGaming по переносу Windows-игр на Linux увенчался 6-й версией. Нумерация у версий *Cedega* необычная. Программа непрерывно развивается, а пользователи покупают подписку на услуги, что позволяет им не только постоянно получать обновленные версии, но и голосовать за игры, поддержку которых они хотели бы получить. Нельзя купить версию 6, продается лишь подписка.

Если TransGaming увеличивает номер версии, это значит, что достигнута некая важная веха в развитии программы. Вехой для версии 6 стала *Oblivion*, популярнейшая игра от Bethesda Softworks, вышедшая в прошлом апреле. Важной вехой *Oblivion* сделала не популярность, а тот факт, что это одна из красивейших и требовательных к ресурсам игр. Она заставляет попотеть даже консоли Xbox 360 и PS3, что уж говорить о нашем скромном Linux-компьютере.



› Чтобы создать эту чудную графику, TransGaming ввела поддержку Shader Model 2.0.

Но чтобы извлечь что-нибудь из *Cedega*, Linux-компьютер должен быть не таким уж скромным. Без хорошей видеокарты от Nvidia не обойтись. Версия 6 заявляет об улучшении поддержки ATI, но, если честно, мы не рекомендуем пользоваться ею, пока не появится нормальный драйвер от ATI/AMD. Инсталлятор программы небольшой – основные файлы загружаются им с сайта TransGaming автоматически. Двоичные пакеты имеются для любого крупного дистрибутива, включая Red Hat, Fedora, Mandriva, Debian и Ubuntu, проблем этот способ инсталляции не вызывает. Установка Windows-игр сводится к простому нажатию непомерно огромной кнопки Install в менеджере игр *Cedega*, выборе профиля игры из выпадающего меню, вставке CD/DVD в привод и нажатию Continue (Продолжить). Прежде чем начать магические процедуры установки, *Cedega* находит инсталляционный файл на диске.

На нашем тестовом компьютере инсталляция *Oblivion* заняла почти полчаса – резкий контраст с восемью минутами для Windows, подорвавший нашу веру в производительность игры. Не было проблем с инсталляцией – и на том спасибо. Любую игру из коллекции можно запустить, выделив ее название и нажав Play. Защиту от копирования *Cedega* берет на себя, используя технологию, лицензированную производителем (одна из причин того, что *Cedega* не свободна), и через несколько секунд появляется панель настройки *Oblivion*. Даже на выверенном ПК с Windows для окончательной настройки игры понадобятся некоторые усилия.

› Запуск игры в окне не влияет на производительность, а вот от Beryl – эффект негативный.



В игре

При низком разрешении 800x600 и среднем уровне качества, *Oblivion* выдавал в среднем 29 кадров в секунду. При том, что наша видеокарта на сегодня не самая мощная (см. Начинка нашего PC, вверху справа), это совсем не плохо. Увеличение разрешения до 1280x760 и включение одного эффекта Bloom уменьшило число кадров в секунду лишь на 10. В общем, игра идет, как полагается, кроме наиболее интенсивных боевых сцен, хотя нам так и не удалось заставить работать в *Cedega* рендеринг высшего качества HDR. Но вот интересный вопрос: с какой скоростью игра идет на Windows? Мы установили Windows и *Oblivion* на тот же компьютер, запустили с теми же настройками и обнаружили, что на Windows скорость увеличилась почти на 40%: до 48 кадров в секунду.

Видеокарты недешевы. Если вам удалось добыть серьезное устройство, необходимое для новейших игр уровня *Oblivion*, то вы не станете мириться со снижением производительности, как бы ни была хороша *Cedega*.

Наш ПК

» AMD Athlon 64 X2 5000+ (Cedega использует только одно ядро)
» 1Гб ОЗУ
» Nvidia 7600GS 256Мб ОЗУ (PCIe)

Но для нерегулярного Linux-игрока *Cedega* – единственная возможность запустить классную игру вроде *Oblivion*. Уже то, что она вообще работает – поразительное достижение. LXF

› Подробнее о кроссплатформенной магии см. стр. 40.

LINUX FORMAT Вердикт

Cedega 6

Разработчик: TransGaming
Сайт: www.transgaming.com
Цена: $5 в месяц (минимальная подписка – три месяца)

Функциональность	9/10
Производительность	7/10
Простота использования	8/10
Документация	9/10

» *Технически впечатляет, неплохо для эпизодического использования, но требует мощного оборудования.*

Рейтинг 8/10

Сравнение

Каждый месяц мы сравниваем тонны ПО – а вы можете отдыхать!

Онлайн-резервирование

Горя желанием избавиться от своих 8-дюймовых дискет, **Крис Хауэллс** оглядел сервисы, позволяющие сохранять данные по сети.



Paul Blachford

Про наш тест...

Тестирование первоначально проводилось на Kubuntu 7.04 Feisty Fawn, с использованием *Mozilla Firefox* (v2.0.0.3) и *Konqueror* (v3.5.6) из KDE. Версия Java была 1.4.2. Важно было учесть, много ли предоставляется пространства для хранилища и какие протоколы передачи данных доступны, так как большее их число дает больше гибкости. Мы также хотели узнать расценки на тот или иной сервис, и существуют ли «поборы» сверх абонентской платы.

Безопасность любой резервной копии – ключевой вопрос. В идеале данные должны шифроваться перед передачей, но скорость сервиса тоже важна: мы использовали быструю (8 Мбит в прямом канале и 1 Мбит в обратном) ADSL-линию, чтобы увидеть, насколько приемлемы скорости. Известно, каким обременительным считают резервирование многие пользователи и администраторы, поэтому регистрировались простота в использовании и качество документации. Наконец, мы учитывали кросс-платформенную совместимость сервиса: хорошо ли он работает с Linux, и есть ли ориентированная на Linux документация.

Наш выбор

Резервирование данных – будь то фотки, привезенные из отпуска тетей Дженни, или платежные ведомости предприятия – важнейшая обязанность, которой часто пренебрегают. При этом оправдываются затратами, сложностями и предположением, что потеря данных – весьма маловероятное событие, которое уж с вами-то никогда не случится.

Однако люди разумные, вроде читателей *Linux Format*, как минимум планируют запустить регулярную процедуру резервирования, а то и прикидывают: не использовать ли онлайн-резервирование для усовершенствования этой операции? Большинство читателей, попавших на наши страницы, знают, что данные при онлайн-резервировании передаются через Интернет на систему хранения провайдера, а не на локально присоединенные физические носители вроде компакт-дисков, внешних жестких дисков или магнитных лент.

Тогда вам уже не надо беспокоиться об утрате резервных копий из-за кражи или пожара, и вы сможете употребить свои ресурсы на другие важные дела. Сюда входит регулярная проверка (вправду ли сохранены те данные, которые вы считаете сохраненными) и планирование операций восстановления данных, если немыслимое событие произойдет. В случае беды следует просто переустановить ОС, получить доступ к онлайн-службе резервирования и восстановить ваши данные.

Что предлагается?

Онлайн-резервирование обычно предлагает три вида методов сохранения и восстановления данных. Во-первых, имеется web-интерфейс, когда отдельные файлы загружаются с помощью HTTP, а множественные файлы/каталоги – с использованием Java-апплета. Второй метод использует программу-клиента, запущенную на вашем компьютере. Лучшие клиенты предлагают всяческие встроенные функции: например, включение задачи резервирования в расписание или опцию использования нескольких наборов файлов для выполнения разных задач резервирования.

Ну и в-третьих, часто можно обратиться к службе, используя FTP, *WebDav* или *rsync*. FTP-клиенты существуют почти для каждой платформы, да и найти подходящие клиенты для *WebDav* или *rsync* также не составит труда. Это – идеальный способ осуществлять резервирование в кросс-платформенном окружении.

Надеемся, что один из сервисов, представленных здесь, подойдет для вас или для вашей организации. Что ж, поглядим!

FilesAnywhere

Вам правда нужны ваши файлы везде?

FilesAnywhere – базирующаяся в США служба, предлагающая разнообразные протоколы передачи данных, равно как и проприетарные Windows-приложения. Как водится для сервисов, базирующихся в США, оплата принимается в американских долларах, и проживающие за пределами США при обращении в службу поддержки могут столкнуться с проблемой часовых поясов, поскольку офисы **FilesAnywhere** открыты только в дневное время США.

Было приятно увидеть Java-апплет для SFTP, то есть вам не надо покидать браузер, чтобы насладиться безопасными FTP-загрузками. Апплет работал неплохо, хотя имел раздражающую привычку открывать окошко для каждого переданного файла, вмешиваясь во все, что бы мы ни делали. Вместо него можно использовать командную строку SFTP-клиента, что придает определенную степень гибкости.

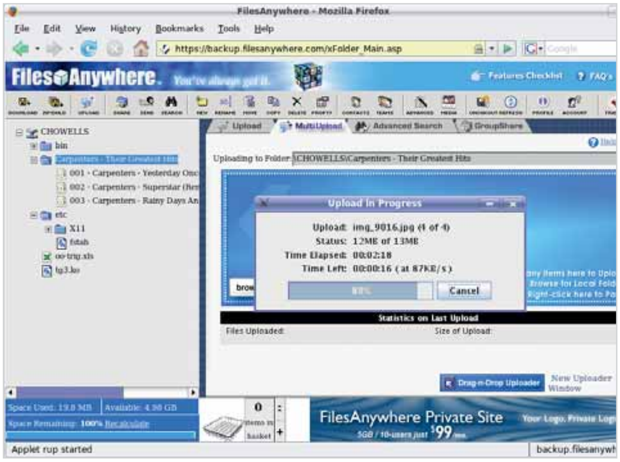
Отличный web-интерфейс дает доступ к файлам через браузер. Он имеет много функций и хорошо оформлен, а Java-опция загрузки означает, что вы можете загружать и целые каталоги, и отдельные файлы, используя HTTP.

Нам понравилась функция Media Gallery, делающая миниатюры изображений для беглого просмотра перед скачиванием. Впечатляет также функция Zip And Download, автоматически сжимающая весь каталог в zip-файл для скачивания: очень удобно, и должно ускорить загрузку.

Существует множество способов совместного пользования файлами: FileShare посылает по электронной почте ссылку на файл для скачивания, GroupShare позволяет разделять файлы с другими пользователями **FilesAnywhere**, и можно даже создать DropBox, чтобы получать файлы от других людей.

Но это еще не все. WebFolders предлагают получить доступ к службе через протокол *WebDav*, и имеется функция Mobile Access, объявленная совместимой с мобильными телефонами BlackBerry и Windows. Функция *WebDav* превосходно работала в *Konqueror* с одноименным KIO slave. Пользоваться ею несложно, и служба оформлена на уровне, с большим выбором доступных протоколов. Так как же мы оценим **FilesAnywhere**? По большому счету, мы ожидаем, что она привлечет продвинутых личностей: документация вся на месте, но в ней мало пошаговых инструкций, а без них некоторым не обойтись.

> «FilesAnywhere дает отличный доступ к файлам из браузера.»



› **Web-интерфейс упрощает загрузку и управление файлами: для загрузки нескольких файлов используется интерфейс Java.**

LINUX FORMAT Вердикт

FilesAnywhere

FilesAnywhere
Сайт: www.filesanywhere.com
Цена: От $3.95 в месяц

» *Впечатляющий сервис со множеством элегантных свойств; браво.*

Рейтинг 8/10

FlipDrive

Щеколда-защелка, другая служба резервирования.

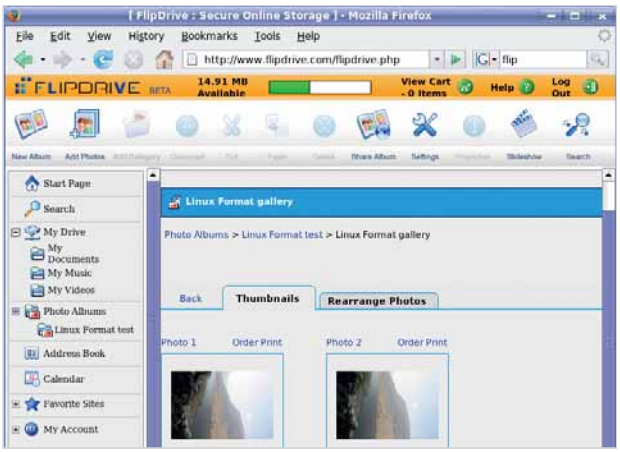
FlipDrive позиционируется скорее как система совместного использования файлов, а не решение для резервирования. Например, можно загрузить большие файлы для общего доступа вместо рассылки по e-mail с риском того, что прикрепленный файл повесит почтовый сервер адресата.

FlipDrive использует систему загрузки файлов, основанную на web-браузере. Интерфейс дает возможность создавать каталоги, загружать новые файлы и, естественно, скачивать ваши файлы. Есть еще и удобная функция поиска по имени файла. Загружать файлы можно двумя методами. Первый – простой метод на базе HTTP, а второй – система 'Power Upload', использующая Java-апплет. Этот апплет позволяет загружать сразу несколько файлов и каталогов. Оба способа предоставляют сведения по количеству переданных данных и времени, оставшемуся до окончания операции.

Система справки эффектна: это видеопособие, сделанное на Macromedia Flash. Есть также функция фотогалереи, хотя она показалась нам не такой интуитивной, как остальной сайт; и, похоже, нельзя добавить в галерею уже загруженные фото.

Совместно использовать файлы с третьими лицами легко, при этом уведомление посылается на электронную почту получателя. Файл можно скачать пять раз за 24 часа – не особенно щедро. Наряду с совместным использованием файлов и резервированием, сайт **FlipDrive** предлагает и другие сервисы, а именно календарь и адресную книгу. Интуитивный и хорошо оформленный, календарь также имеет функцию извещения по электронной почте; это удобно, если вы постоянно забываете о чьем-нибудь дне рождения.

> «FlipDrive — система и общего доступа к файлам, и резервирования.»



› **FlipDrive предлагает фотогалереи, к которым можно дать открытый доступ.**

LINUX FORMAT Вердикт

FlipDrive

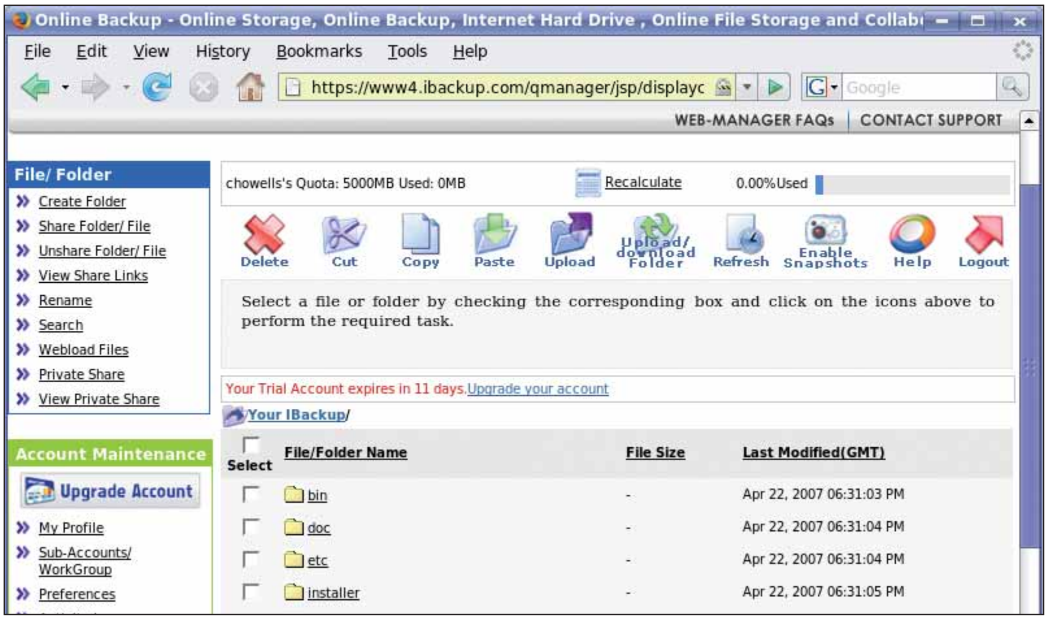
FlipDrive
Сайт: www.flipdrive.com
Цена: От $4.95 в месяц

» *Хорошо оформленное приложение, с хорошим базовым сервисом совместного использования файлов и резервирования.*

Рейтинг 8/10

IBackup

Нет, это не очередной продукт Apple...



IBackup – это сервис от Pro Softnet Corporation из Калифорнии. Он предлагает три вида учетных записей. План Economy включает только хранение и имеет самую низкую цену, план Enhanced – продукт средней категории, а план Workgroup предлагает большую часть функций по самой высокой цене. Некоторые возможности, стандартные для плана Workgroup, в учетных записях Economy и Enhanced доступны как необязательные дополнения.

Наше первое впечатление было очень позитивным: всеобъемлющий диапазон предлагаемых протоколов – просто блеск! Регистрация, однако, разочаровала нас приглашением скачать приложения для резервирования **IBackup** для Microsoft Windows, тогда как наш браузер был *Mozilla Firefox* под Linux.

Вернувшись на главную страницу, мы смогли задействовать основной сервис **IBackup**, состоящий из простой системы загрузки файлов через web. Сервис работает адекватно, но по-настоящему подходит только тем, у кого есть несколько файлов, не нуждающихся в особо частом резервировании. Интерфейс довольно интуитивен и обеспечивает нормальные средства управления файлами, такие как переименование и перемещение.

Облом с папками

Если вам нужно сделать резервную копию большого числа файлов, вас порадует, что можно резервировать папки целиком, используя WebDav-подобную систему. После ввода имени пользователя и пароля мы получили ошибку HTTP 403 'Permission denied'. Мы было не сильно огорчились, поскольку предыдущая страница предупреждала, что перед использованием этой функции следует установить расширение *Openwebfolder* для *Firefox*. Но увы, оно не помогло.

Затем мы перешли на *Konqueror*, где есть хорошая встроенная поддержка *WebDav*. Это также не дало результата, **IBackup** выдавал сообщение 'Browser not supported'.

Для пользователей Windows **IBackup** предлагает функцию сетевого диска под названием *IDrive* – увесистое 10-МБ приложение. Оно позволяет обращаться к сервису как к обычному диску под Windows. Мы надеялись, что эта функция основана на SMB-протоколе, обычно используемом в сетевых дисках под Windows – он хорошо поддерживается в Linux; и просчитались.

Мы проглядели весь сайт, чтобы побольше узнать о совместимости с Linux. К нашему удивлению, оказалось, что поддерживается *rsync* – довольно умный протокол, старающийся минимизировать количество передаваемых данных; неплохое решение при резервировании сотен мегабайт ежедневно. Он порадовал нас отличной работой. **IBackup** позволяет запускать *rsync* через *Stunnel*, ради шифрования данных.

Совместное использование файлов

Чтобы дать другим доступ к вашим данным, предлагаются различные функции. Есть функция Private Share, позволяющая вам совместно использовать файлы с другим пользователем **IBackup**. Вдобавок функция Sub-Account открывает файлы для третьих лиц, что, однако, требует дополнительной платы $5 в месяц за каждый Sub-Account. Поскольку третьим лицам дается возможность изменять файлы, предусмотрена блокировка: изменения, сделанные третьей стороной, не перезаписываются изменениями, сделанными другой третьей стороной.

Одна уникальная возможность нас заинтриговала. Она касается выполнения первоначальной резервной копии, с использованием внешнего жесткого диска или сервера, который затем переправляется в **IBackup** и вносится в систему хранения компании. Последующие резервирования, производимые приложениями Windows, Mac OS X или *rsync*, передают только те данные, что были изменены.

IBackup также предлагает проприетарное Windows-приложение для создания резервных копий. Оно полнофункционально, и включает массу интересных деталей: например, экономит трафик и время путем передачи только изменившейся части файлов. Программа-клиент также дает специальную поддержку для резервирования баз данных *Microsoft Exchange* и *Microsoft SQL Server*. А раз уж имеется клиент для Mac OS X, было бы очень приятно увидеть в будущем клиента и для Linux.

> **Web-интерфейс IBackup толково разработан.**

IBackup хранит свои данные на серверах, находящихся в США. В общем, на нас произвела сильное впечатление техническая сторона сервиса **IBackup**, но способ его предоставления впечатляет куда меньше: он вряд ли подойдет технически неопытным пользователям. Хотелось бы, чтобы информация о функциях, доступных для учетной записи, была ясно изложена на сайте – например, чтобы узнать о возможности доступа к серверам **Ibackup** через *rsync*, нам пришлось порыться в документации.

LINUX FORMAT Вердикт

IBackup

Сайт: www.ibackup.com
Цена: От $9.95 в месяц

» *Хороший диапазон протоколов, но сервис не так отполирован, как остальные.*

Рейтинг 7/10

IceBackup

Выручит ли вас это ледяное решение?

На весьма профессионально выглядящем сайте **IceBackup** специально оговорено, что их сервис поддерживает все дистрибутивы Linux. Впечатляет сам набор поддерживаемых операционных систем, включающий Windows, Mac OS X, Solaris и Novell NetWare.

Сначала нам показалось странным, что размер клиента для Linux такой большой, примерно 30 МБ. Он содержит как консольную версию, так и версию для X11.

Мы честно следовали всем (довольно скверным) инструкциям, но нам, тем не менее, не удалось разрешить странную ошибку Java, якобы связанную с файлом, не включенным в дистрибутив. Провозившись с этой проблемой несколько часов, мы пришли к выводу, что совместимость **IceBackup** с Linux не так хороша, как обещано.

Тогда мы перешли к консольной версии. Следуя опять-таки неадекватным инструкциям, мы не сумели исправить ошибку 'Configuration File NOT Created'. К сожалению, других сообщений не поступило – а было бы весьма интересно узнать, почему же все-таки не создался файл конфигурации.

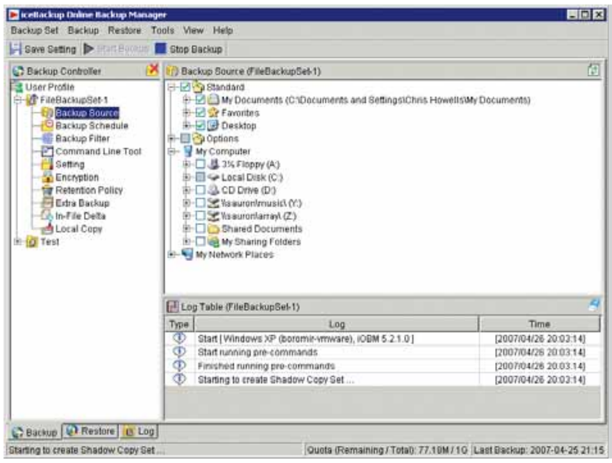
Клиент для Windows поживает существенно лучше: он набит функциями, осуществляющими различные настройки резервирования и планирование таковых; зато он не отображает скорость передачи данных, так что очень сложно оценить процент завершения и понять, сколько еще осталось ждать, а это раздражает.

Впечатляет то, что клиент позволяет непосредственно подключаться к серверам *MySQL* и Oracle для их резервирования – это существенно упрощает задачу создания резервных копий серверов баз данных, поскольку нет необходимости проводить собственное резервирование при помощи Cron и *mysqldump*.

Отчеты о состоянии, где перечисляются каталоги, помещенные в резервную копию, приходят по электронной почте, независимо от успешного либо неудачного завершения операция. Это приятная деталь, но опять раздражает отсутствие информации о том, сколько времени длилось резервирование, и какие скорости передачи при этом достигались.

Клиент довольно прост в использовании, но явно нацелен на тех, кому нужно серьезное решение для резервирования. Из-за сложности используемых концепций планирования и файловых наборов, для новичков это неподходящий выбор.

> «Полный список поддерживаемых ОС впечатляет.»



› Клиент IceBackup для Windows создает log-файл, и вы можете узнать, что происходит.

Вердикт

IceBackup

Сайт: www.icebackup.net
Цена: От $14 в месяц

» Продукт для Linux и техническая поддержка сильно разочаровывают.

Рейтинг 3/10

XDrive

Это не драйв, это XDrive.

Если вы возжелаете воспользоваться **XDrive**, придется завести учетную запись AOL. Ясно, насколько сильно ориентирован на США этот сервис: требуется вводить телефонный номер и почтовый индекс в формате США, а где ж их взять, коли у вас их нет.

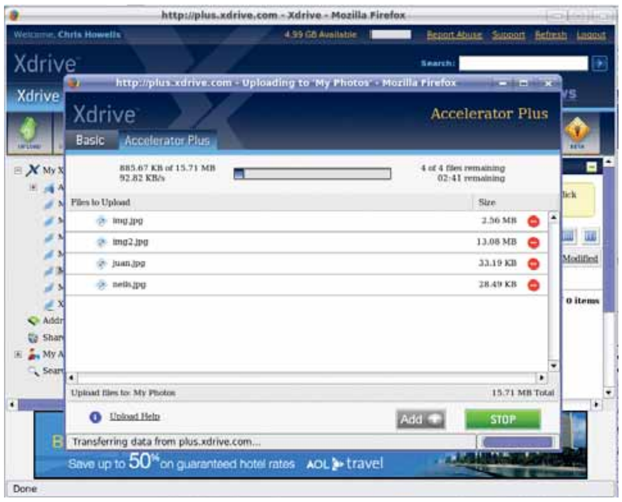
XDrive щедро дарит 5 ГБ для бесплатного хранения, а 50 ГБ стоят всего $9.95. Халява – это святое, и **XDrive** – единственный подобный сервис, предлагающий бесплатно значительное место для хранения. После входа на сайт вам предоставляется меню Launchpad, предлагающее выбор: либо открыть виртуальный жесткий диск, либо запустить Xdrive-шоу. Последнее – это новаторский метод создания слайд-шоу из рисунков и музыки, идеально работающий с *Firefox* – он основан на Macromedia Flash.

Загрузка файлов может осуществляться двумя способами. Первый, Basic, использует простую HTTP-передачу. Он прилично работает для отдельных файлов, и даже предоставляет информацию о ходе загрузки. Второй метод называется Accelerator Plus; он предоставляет хорошие скорости – наша ADSL-загрузка добралась до 100 Кб/сек. Главный его козырь – способность загружать целые каталоги за один раз. После загрузки файлов **XDrive** работает весьма интеллектуально, обеспечивая предпросмотр для больших рисунков. Также имеется функция поиска.

Интересна обнаруженная нами функция закладок, где можно хранить Закладки/Избранное web-браузера. В настоящий момент, однако, поддерживается только импортирование закладок *Internet Explorer*.

Сервис корректно работал с *Firefox*, а вот *Konqueror*, к сожалению, отказал в защищенном входе.

> «XDrive щедро дарит 5 ГБ на бесплатное хранение.»



› Accelerator Plus обеспечивает высокую скорость загрузки.

Вердикт

XDrive

Сайт: www.xdrive.com
Цена: бесплатен в базовом варианте

» Реализовано несколько очень удобных домашним пользователям функций.

Рейтинг 7/10

Rsync.net

Предлагая больше, чем просто *rsync*.

Строгий сайт сразу переходит к техническим деталям – на фоне назойливо-рекламных сайтов конкурентов это бодрит – и мы были мгновенно очарованы предложениями сервиса **Rsync.net**. Столь же приятно смотрится документ об условиях обслуживания, написанный для людей без юридической практики: безо всякой там неудобочитаемой зауми, которой часто злоупотребляют подобные документы.

Rsync.net имеет три центра данных: два в США, один в Швейцарии. В этом году также планируется расшириться в Индию и на Тихоокеанское побережье. Есть даже опция выбора центра данных, где бы вы хотели хранить ваши файлы – европейцы, скорее всего, предпочтут Швейцарию, чтобы снизить время ожидания. Другая интересная функция – «геодублированная файловая система», которая автоматически воспроизводит данные на другом месте. Это снижает вероятность причинения вреда вашим данным стихийными бедствиями типа наводнений.

Rsync.net не снабжает программами для доступа к сервису, но рекомендует пользоваться решениями третьих фирм. Следовательно, вашим вкусам угодят в любом случае, будь вы поклонником хоть консольных FTP-клиентов, хоть графических клиентов *rsync*.

Rsync.net нацелен скорее на опытных пользователей Unix, чем на желающих просто сохранить пару-тройку файлов, и предлагает ряд протоколов, большая часть которых поддерживает шифрование. Проще всех – старый добрый FTP и *WebDav*. Для продвинутых пользователей предлагается множество протоколов туннелирования посредством SSH, включая *rsync*, SFTP и SCP. Для тех, кто хочет поиграть со следующим поколением интернет-адресации, имеется IPv6.

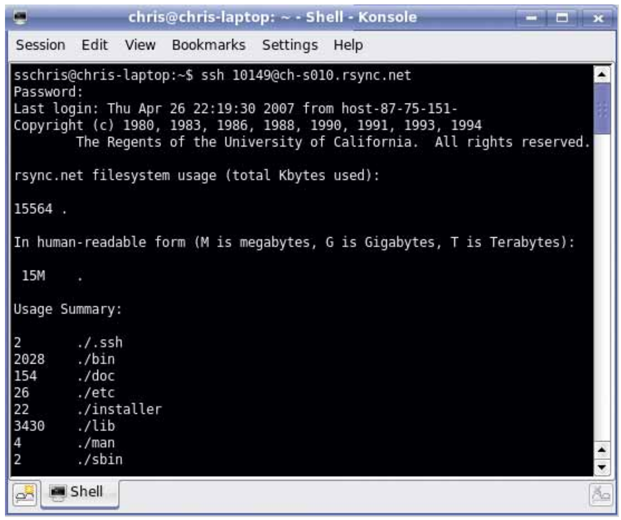
Друзья администратора

Rsync.net предлагает несколько приятных функций, включая возможность отображать учетную запись *WebDav* как диск под Windows, то есть служба будет выглядеть под Windows просто как новое имя диска в дереве. Пользователи Unix могут с помощью SSHFS создавать виртуальную файловую систему. Поясним, что SSHFS передает данные по протоколу SSH, а выглядит как обычная локально примонтированная файловая система. Для настройки SSHFS под Linux и FreeBSD имеются инструкции. Не забыта и OS X для Apple Mac: даны инструкции по настройке *Finder* для обеспечения прямого доступа к сервису через рабочий стол OS X.

Для создания резервных копий компания рекомендует утилиту под названием *Duplicity* (**http://duplicity.nongnu.org**). *Duplicity* применяет *GNU Privacy Guard*, *GnuPG*, для шифрования данных с использованием Pretty Good Privacy (PGP). Для эффективного управления передачами сжатых резервных копий служит *librsync*, при этом гарантировано, что при последующих актах резервирования будут передаваться только изменения файлов.

Как и следовало ожидать, **Rsync.net** дает хорошие инструкции по использованию утилиты *rsync* под Windows, Mac и Linux, приводя прямые примеры требуемых настроек. Даны также ссылки на подходящие версии *rsync* для Windows и OS X. Еще нам понравились обещания, касающиеся поддержки – она будет осуществляться инженерами, а не техниками поддержки первого уровня, способными разве что читать по бумажке. Каждый, кому хоть раз приходилось тратить время на телефонные разговоры с технической поддержкой, без сомнения, ощутит признательность. Документация на сайте тоже весьма обширная, хотя и нацелена на технически грамотных людей.

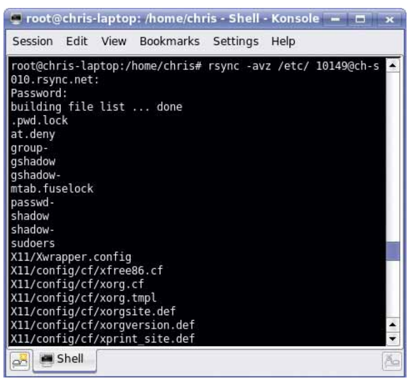
Помимо технической части, **Rsync.net** предлагает различные денежные премии за улучшение кодов сообществом – в частности, кода *Duplicity*, выпущенной под GPL. Также приятно видеть, что имеются скидки на пользование этим сервисом для разработчиков открытых приложений.



› **Размеры файлов и данные об использовании учетной записи можно узнать через SSH.**

> «SSHFS представляет Rsync.net как виртуальную файловую систему.»

В целом, качество **Rsync.net** произвело на нас отличное впечатление: это гибкая система, которой можно пользоваться из целиком открытой среды. Нельзя сказать, что **Rsync.net** отличается простотой: сервис – не для новичков, но это простительно, поскольку не они основные потребители **Rsync.net**.



› ***Rsync* – просто инструмент для регулярного резервного копирования серверов.**

Освоим жаргон

» ***rsync*** – утилита для эффективного обмена данных между сайтами. Важно, что пересылаются только различия между разными версиями файла, и если файл был изменен совсем немного, то следующее резервирование выполняется гораздо быстрее.

» **SSH** обозначает Secure SHell, удаленную систему доступа, широко используемую при администрировании Unix-серверов. В отличие от раннего Telnet, данные здесь шифруются.

» **SFTP** – SSH-протокол передачи файлов, который работает путем туннелирования данных передаваемых файлов через SSH-соединение, обеспечивая невозможность перехвата файлов.

» **SCP** (Secure Copy Protocol) – другой способ передачи файлов через SSH-соединение.

LINUX FORMAT Вердикт

rsync.net

Сайт: www.rsync.net
Цена: От $1.60 в месяц

» *Отличный сервис, адресованный опытным пользователям и профессионалам.*

Рейтинг 8/10

Службы онлайн-резервирования

Вердикт

Rsync.net 8/10

Мы рассмотрели несколько сервисов резервирования, которые четко направлены на две разных категории: тем, кто просто хочет изредка сбрасывать куда-нибудь несколько файлов, возможно, для совместного использования с другими, и тем, кому нужно регулярно резервировать большие объемы данных.

Если вы входите в первую категорию, можете положиться на сервисы, предоставляемые **XDrive**, **FlipDrive** и **FilesAnywhere**. Пользователей, желающих сохранять большие массивы данных на более регулярной основе, заинтересуют сервисы от **IBackup**, **Rsync.net** и **IceBackup**.

Наряду с требуемым объемом дискового пространства для хранения, еще один важный фактор, который следует принять во внимание при выборе сервиса онлайн-резервирования – объем передаваемых данных.



› **FilesAnywhere хорошо выполняет операции с различными типами файлов, создавая миниатюры для рисунков.**

Rsync.net явно указывает, что объем практически не ограничен, а два других сервиса, **XDrive** и **FlipDrive**, придерживаются политики «честного использования».

По правде сказать, нас обеспокоила та часть пользовательского соглашения **XDrive** и **FlipDrive**, которая, не оговаривая точных значений лимита, грозит прекращением обслуживания за использование «значительно больших объемов, чем остальные потребители сервиса».

Две службы понравились нам значительно больше остальных. Первой стал **Rsync.net**, отличающийся серьезным подходом, что вносит приятное разнообразие. Мы считаем, что это отличное решение в финансовом плане – хранение 1 ТБ стоит $640 в месяц при неограниченных передачах данных – хотя он потерял очки из-за сложности в использовании. Также приятно видеть систему загрузки, основанную на web, и кое-какую документацию, ориентированную на менее опытных пользователей.

Второй оказался **FilesAnywhere**. Им легко пользоваться, и он хорошо документирован – то самое, что нужно менее опытному пользователю. Мы также обнаружили, что **FilesAnywhere** предлагает отличные возможности для совместного использования файлов с другими пользователями и хороший выбор протоколов. Цены повыше, чем у других сервисов, но вполне приемлемы. Если цена для вас – главная забота, рекомендуем обратиться к куда более дешевому **FlipDrive**.



› **Поддержка и сервис Rsync.net впечатляют с любой стороны. Можете выбрать, где хранить ваши данные: в США или в Европе.**

Весьма неутешителен тот факт, что немало онлайн-сервисов, протестированных нами, на практике оказались несовместимыми с Linux из-за использования проприетарных Windows-клиентов. Однако в будущем следует присмотреться к сервису **Mozy** (www.mozy.com), чьи создатели обещают родной Linux-клиент уже в этом году. LXF

Обратная

Пользуетесь ли вы онлайновыми службами резервирования? Нам будет интересно узнать, какого типа онлайновую систему вы используете, и почему. Поделитесь вашим мнением и опытом на letters@linuxformat.ru.

Таблица характеристик

Сервис	Доступное пространство	Цена	HTTP	WebDav	FTP	rsync	Проприетарный клиент	Работа с Konqueror	Работа с Firefox	Совместное использование файлов	Фото-галерея	Скорость	Стоимость передачи данных
Files Anywhere	5ГБ-500ГБ	100 МБ $4/месяц, 50 ГБ $70/ месяц	✓	☒	✓	☒	✓ Только Windows	Плохо	Хорошо	✓	☒	Хорошо	Не указано
FlipDrive	25МБ-100ГБ	25 МБ $5/месяц, 100 ГБ $20/ месяц	✓	☒	☒	☒	☒	Плохо	Хорошо	✓	✓	Отлично	Политика честного исполь-зования
IBackup	5ГБ-300ГБ	5 ГБ $10/месяц, 300 ГБ $300/ месяц	✓	✓	✓	✓	☒	Хорошо	Хорошо	✓	☒	Отлично	Не указано
IceBackup	500МБ-100ГБ+	500 МБ $14/ месяц, 100 ГБ $280/месяц + $50 за установку	☒	☒	☒	☒	✓ Windows/Linux/ OS X	Не работает	Не работает	☒	☒	Хорошо	Не указано
Rsync.net	1ГБ-1ТБ+	1 ГБ $1.6 /месяц, 1000 ГБ $640/ месяц	☒	✓	☒	✓	☒	Не работает	Не работает	☒	☒	Отлично	Без ограничений
XDrive	5ГБ-50ГБ	5 ГБ бесплатно, 50 ГБ $10/ месяц	✓	☒	☒	☒	✓ Только Windows	Плохо	Приемлемо	✓	✓	Хорошо	Политика честного исполь-зования

Distrowatch

Ежемесячная сводка новостей дистрибутивов Linux



ЛАДИСЛАВ БОДНАР
основатель, начальник, редактор и сотрудник DistroWatch.com.

Дружелюбие Linux

Помните ли вы, как устанавливали Linux в первый раз? Вы вставляли дискету и перезагружались, затем вводили таинственные команды для разбиения диска на разделы, установки ядра и приложений и настройки загрузчика. После этого часами настраивали X Window, модем, звуковую карту, принтер... Если вам везло и ваши устройства были дружелюбны к Linux, то, потратив все выходные, вы получали пригодную к использованию систему.

К 2007 г. многое изменилось. Ведь каждый дистрибутив Linux теперь предлагает простой установщик в стиле «щелкай и выбирай», с практически идеальным определением устройств и рабочим столом с тысячами приложений. Обретенное удобство привлекло много новых пользователей, ранее сторонившихся не в меру технической системы.

Что же дальше? Успех небольшого числа малых дистрибутивов (вроде Linux Mint, Sabayon Linux и PCLinuxOS) говорит о том, что новичкам нужны три вещи: преднастроенные современные видеокарты и драйвера беспроводных сетей, модули браузеров, позволяющие без проблем воспроизводить популярные сетевые медиа-форматы, и поддержка несвободных аудио- и видеоформатов, вроде MP3.

Держимся за принцип?

Из основных дистрибутивов Linux, только Ubuntu и Mandriva расценивают это как важные вещи. Остальные продолжают (в той или иной мере) придерживаться «правильного» подхода, не поступаясь принципами свободного ПО. Хотя это и заслуживает уважения, но в битве за рынок – по крайней мере для домашнего пользователя – эти дистрибутивы теряют шансы переманить в Linux побольше людей.

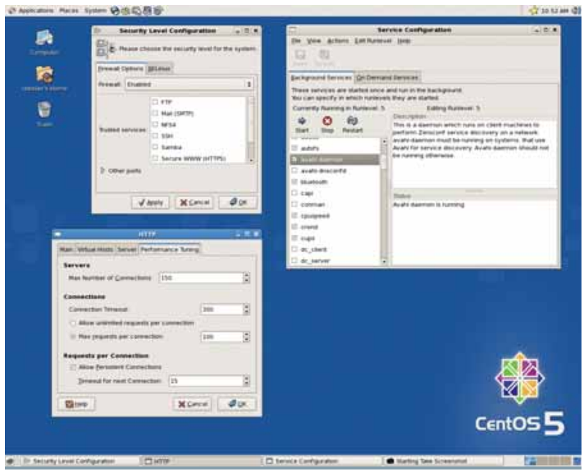
ladislav.bodnar@futurenet.co.uk

Топ-клон

CentOS 5.0 Лидер среди ОС, перекомпилированных из перекомпиляций Red Hat Enterprise Linux.

Когда Red Hat прекратила выпуск свободно доступного Red Hat Linux, разветвив его на Fedora Core и Red Hat Enterprise Linux, многие были разгневаны. «Нам не нужна поддержка!» – восклицали одни, обсуждая дороговизну корпоративного решения. «Fedora недостаточна хороша!» – жаловались другие на новый непротестированный дистрибутив с малым временем жизни. В 2003 г., когда Red Hat была самым популярным Linux-дистрибутивом в мире, многие были не готовы к такому шагу компании из Северной Каролины.

К счастью, причиной недовольства была ОС с открытым кодом, а не лицензируемый, патентованный проприетарный продукт. Первым, кто извлек из этого выгоду, был Джон Моррис [John Morris], системный администратор публичной библиотеки в Дериддере, США. Библиотека не могла заплатить за лицензию за корпоративное решение Red Hat с поддержкой для каждого рабочего места, поэтому Моррис пересобрал исходные тексты, свободно доступные на сервере Red Hat, и выпустил свой продукт под именем White Box Enterprise Linux.



› Как и его знаменитый предок, CentOS предоставляет несколько собственных инструментов администрирования сервера.

За ним последовали схожие проекты. Работа была не очень сложной; она включала удаление всех упоминаний о товарных знаках Red Hat, компиляцию исходного кода в двоичные пакеты и размещение результатов на загрузочном носителе для простой установки. Приобретая известность, эти клоны RHEL различались по уровню коммерциализации и изменениям, но имели одну общую черту: они предлагали минимум пять лет обновлений безопасности и исправлений, как и «настоящий» Red Hat Enterprise Linux.

Сглаживание углов

Эти проекты были спорными. Хотя перекомпиляция исходного кода Red Hat не была противозаконной, некоторым было трудно принять факт, что любой владелец компилятора мог легко создать ОС, в разработку которой были вложены тысячи человеко-часов Red Hat. Проекты-клоны отвечали, что они просто предоставляют сервис для малого бизнеса, которому не по карману Red Hat Enterprise Linux. Сама Red Hat не жаловалась, утверждая, что ее настоящая ценность – техническая поддержка, а не сам продукт, но юристы компании заставили часть этих проектов удалить любые ссылки на Red Hat с официальных web-сайтов.

С годами некоторые из родившихся проектов воспользовались доступностью исходного кода для продажи своих клонов Red Hat с недорогой поддержкой. Позднее, Oracle, гигант баз данных, сделала большую заявку на этом рынке, объявив о доступности «улучшенного» клона RHEL с программой поддержки под названием Oracle Unbreakable Linux.

Однако наиболее популярным среди клонов RHEL остается проект CentOS. Это полностью свободный дистрибутив сообщества с довольно быстрыми обновлениями безопасности для все трех его поддерживаемых версий. В настоящее время CentOS также предлагается некоторыми лидерами web-хостинга как альтернатива продуктам Red Hat уровня предприятия, и он также был одним из первых проектов, успешно пересобравших последний релиз RHEL 5 в CentOS 5.0. Не будучи приемлемым для компаний с сотнями сложных компьютерных систем, CentOS и подобные клоны могут помочь малому бизнесу, нуждающемуся в стабильной ОС с долгосрочными обновлениями безопасности.

www.centos.org

Освежающе классный

Linux Mint 2.2 Кодовое имя Bianca, мультимедиа-вариант Ubuntu, который «просто работает».

Linux Mint – один из сюрпризов 2007 года. Проект затевался просто как web-сайт для новичков Linux, но его основатель, Клеман Лефевр [Clement Lefebvre], скоро определил общую жалобу тех, кто первый раз попробовал Linux: отсутствие поддержки популярных медиаформатов в большинстве дистрибутивов. В результате он задался целью переработки Ubuntu в дистрибутив, который включал бы все стандартные медиакодеки. Позднее он добавил дополнительный глянец, типа собственных тем, поддержки стандартных беспроводных сетевых карт и полезных модулей расширения для браузеров. Linux Mint моментально стал хитом.

Последняя версия, 'Bianca', основана на Ubuntu 6.10. Подобно Ubuntu, она поставляется в виде LiveCD с возможностью установки на жесткий диск и предпочитает Gnome, но на этом сходство заканчивается. Собственный набор приложений содержит несколько занятных элементов, вроде медиа-проигрывателя Amarok (KDE-приложение), тогда как другие приложения, типа *OpenOffice.org*, представлены более свежими, чем в Ubuntu, версиями.

Но более всего притягивает новых пользователей в Linux Mint поддержка MP3-музыки, зашифрованных DVD и другого популярного медиа-контента «из коробки». Браузер *Firefox* также содержит длинный список дополнений, таких как Flash, Java, *QuickTime*, *RealPlayer* и *Windows Media Player*. Утилита под названием *MintConfig* поможет с установкой драйверов беспроводной сети, монтированием разделов или настройкой управления питанием. MintDesktop проведет мигрантов с Windows через другие головоломные задачи.



› У Linux Mint милый рабочий стол, с улучшенным меню «Старт».

Linux Mint [mint – «мята», *англ.*] – освежающее явление на сцене дистрибутивов. Если вы новичок в Linux или у вас есть друзья или родственники, которые не очень-то ладят с компьютерами, то Linux Mint может быть прекрасной ОС для старта.

www.linuxmint.com

Fedora без Core

Fedora 7 В этом тяжеловесном дистрибутиве больше приложений и LiveCD с возможностью установки.

В соответствии со старой системой нумерации версий Red Hat, эта версия ее свободного дистрибутива должна была быть версией «точка-ноль». В отличие от своего предшественника, стабилизационного релиза, предполагавшего формирование прочной основы для Red Hat Enterprise Linux 5, Fedora 7 неприкрыто экспериментальна. Просматривая информацию о выпуске, более чем ясно видишь, что как только RHEL 5 вышел в свет, команда разработчиков Fedora получила больше свободы для воплощения эксцентричных идей и экспериментальных функций в новом продукте.

Даже беглый взгляд на сервер загрузок Fedora говорит о масштабе изменений, представленных в Fedora 7. Старые каталоги 'core' и 'extras' слиты в огромный репозиторий из более чем 7000 программных пакетов. Он содержит последнее ядро 2.6.20, последний релиз *X.org 7.2*, три современных графических окружения (Gnome 2.18, KDE 3.5.6 и *Xfce 4.4*), стремительно набирающие популярность технологии трехмерного рабочего стола *Beryl* и *Compiz*, последний *OpenOffice.org 2.2*, виртуализацию *Xen* и стандартный набор настольных, серверных, мультимедийных приложений и средств разработки.



› Впечатляющий дизайн предназначен для привлечения большего числа настольных пользователей.

Другая важная функция – официальный LiveCD, способный также служить и установочным диском. LiveCD поставляется в Gnome- и KDE-вариантах, и содержит установщик Anaconda. Пользователям со скоростным интернет-соединением несомненно понравится эта опция; вместо загрузки больших образов DVD, они могут просто вставить LiveCD и установить дополнительные пакеты из сети.

Fedora 7 содержит несколько более тонких улучшений, вроде ускоренных загрузки и установки пакетов, но больше всего, несомненно, запомнятся великое объединение программных пакетов и удаление из имени приставки «Core».

www.fedoraproject.org

Клоны RHEL

Кроме CentOS, варианты Red Hat Enterprise Linux предлагают несколько других проектов. Большинство из этих дистрибутивов доступны для свободной загрузки, хотя некоторые из начинающих, ради выгодного бизнеса, предлагают установочные диски и обновления безопасности за небольшую ежемесячную плату. Ниже в таблице приведен полный список дистрибутивов, специализирующихся в пересборке RHEL. На момент написания, только CentOS и StartCom выпустили версии, основанные на RHEL 5, но остальные, похоже, уже готовы к этому.

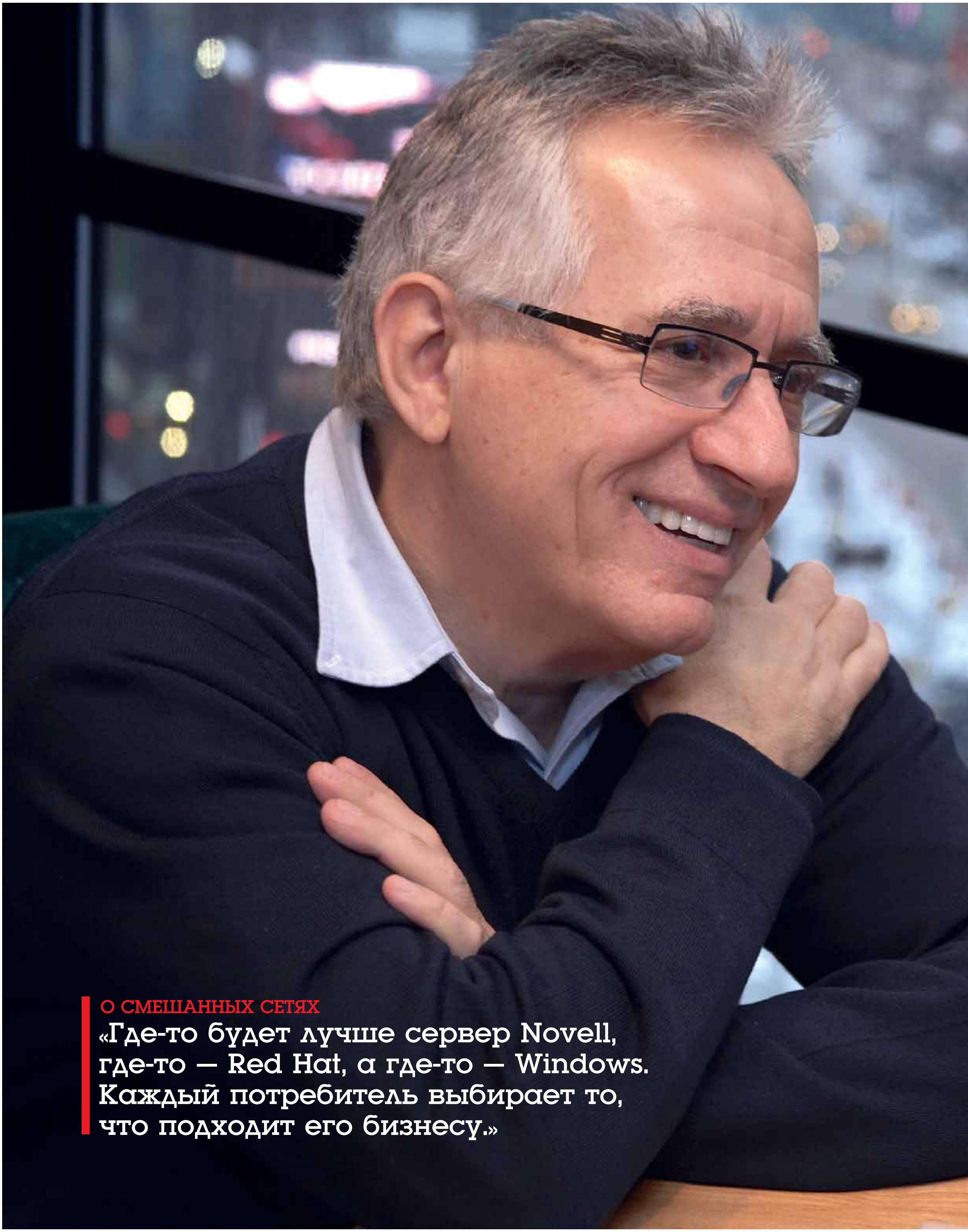
Дистрибутив	Описание
CentOS	Чистая пересборка RHEL без значительных улучшений; проект свободного сообщества.
StartCom Enterprise Linux	Предоставляет чистую пересборку и мультимедийное издание; бесплатен.
Scientific Linux	Пересборка RHEL, дополненная научными и другими приложениями.
Lineox	Полукоммерческий проект, основанный в Финляндии (www.lineox.net); чистая пересборка; очень быстрые обновления.
Pie Box Enterprise Linux	Коммерческий проект, базирующийся в Великобритании (www.piebox.org); предлагает медиа и обновления за небольшую месячную плату.
Oracle Enterprise Linux	Пересборка CentOS с различными улучшениями; предлагает дешевую поддержку (www.oracle.com/technologies/linux).
Fermi Linux	Пересобранный Scientific Linux, усиленный дополнительными функциями безопасности; бесплатный.
White Box Enterprise Linux	Чистая пересборка RHEL; первый созданный клон RHEL; бесплатный.
X/OS Linux	Голландский проект; чистая пересборка RHEL; бесплатный.
Black Orion Enterprise Linux	Коммерческий проект Linux (www.blackorion.com); чистая пересборка RHEL.

Хит-парад дистрибутивов

10 самых посещаемых страниц на DistroWatch.com с 4 апреля по 3 мая 2007 г. (среднее число визитов в день)

Дистрибутив	Число визитов	
1 Ubuntu	2,912	‹›
2 SUSE	2,065	↑
3 PCLinuxOS	1,522	↓
4 Fedora	1,497	‹›
5 SimplyMepis	1,232	↑
6 Debian GNU/Linux	1,107	↑
7 Mandriva	995	‹›
8 Sabayon Linux	906	↓
9 Damn Small Linux	801	↑
10 Mint	752	↓

» DistroWatch.com определяет популярность дистрибутивов, основываясь на числе визитов на страницу каждого дистрибутива. Хотя это не дает представление о действительном числе установок, но показывает, какие дистрибутивы более востребованы за данный промежуток времени.



О СМЕШАННЫХ СЕТЯХ

«Где-то будет лучше сервер Novell, где-то — Red Hat, а где-то — Windows. Каждый потребитель выбирает то, что подходит его бизнесу.»

Photography: Jena Cumbo



Нечистые мысли

Андреас Типальдос считает, что не все предприятия хотят или могут ограничиться чистым Linux. Если Linux – как и его фирма, Xandros – хочет преуспеть, сказал он ***Linux Format***, придется признать, что наш мир – это мир смешанных платформ.

Вы, вероятно, знаете Xandros как фирму, которая в 2001 г. купила Corel Linux, основанный на Debian. Фактически, ее инженеры по-прежнему создают настольный дистрибутив, собирающий немало похвал за свое исключительное дружелюбие. Сама же Xandros критикуется за невнимание к сообществу и закрытость кода некоторых программ. Но недавно глава компании Xandros, Андреас Типальдос, привел ее в корпоративные воды, начав производство серверных и бизнес-дистрибутивов, а также инструментов для администраторов, работавших Windows. Типальдос, ветеран технологии и инвестор, основавший Ximian и CodeWeavers, не оправдывается за то, что ориентируется на пользователя и концентрирует внимание на интуитивном графическом интерфейсе, «к великому огорчению асов командной строки», как говорит он. Возможно, это не соответствует вашему представлению о чистоте свободного ПО, но Типальдос отстаивает ту точку зрения, что подобный прагматичный подход – лучший способ перехода на Linux. **Ник Вейч** встретился с ним, чтобы узнать больше.

Linux Format: У вас есть какие-либо выводы по поводу сделки между Novell и Microsoft?

Андреас Типальдос: Я думаю, частично они заключаются в том, что Microsoft не в восторге от Linux, это очевидно. С другой стороны, они касаются минимизации урона. Если вы – пользователь…

LXF: Полагаю, им известно, что практически все их клиенты на рынке серверов пользуются тем или иным Linux.

AT: Видите ли, я прошел весь путь проприетарных платформ, когда были Wang, DEC, Data General, IBM Series 1, а затем мы перешли на открытые системы – Unix и Linux – и даже на другие платформы в прочих областях, например, в базах данных. И я считаю, что реальность такова: никто не сумеет заставить пользователя принять стандарт. У Goldman Sachs и других моих клиентов, в центрах обработки данных используется Red Hat, а для идентификации – *Active Directory*, и это факт. Мне говорят: «Знаете, они должны работать вместе: вам не удастся заставить нас ограничиться чем-то одним». Так что я думаю, частью [этого союза] является то, что Microsoft осознает или принимает тот факт, что Linux никуда не денется.

LXF: Вот интересно: вопрос интеллектуальной собственности (ИС) висит над Linux. Наступит ли день, когда он исчезнет?

AT: Я мыслю реалистично: по мере того, как время идет, и Linux понемногу превращается в стандарт, кто будет его отменять? То же случилось и с Windows: спустя »

» какое-то время она стала стандартом, несмотря на все эти проблемы с монополизмом и прочим. И никуда ее не деть, иначе рухнет мировая экономика. Чем дальше, тем более спорным становится этот вопрос.

И есть еще один момент: предположим, что вы – Microsoft. Вы предъявите иск Goldman Sachs, чтобы они прекратили пользоваться Linux из-за проблемы с ИС? Вы бы выиграли битву, но проиграли войну. Я думаю, большая опасность исходит от SCO и иже с ними.

LXF: Потому что им больше терять нечего?

AT: Именно. Не представляю, чтобы Microsoft предъявила иск крупным клиентам, хотя они и пытаются выиграть PR-битву за ИС. Потому что, как вы сказали, им есть, что терять: в конце концов, потребитель обидится.

Реальность такова, что это – гетерогенный мир, мир смешанных платформ, у потребителя свои заботы, и всякий, кто не принимает их в расчет, наносит вред самому себе. Война проприетарного и открытого, которая длится уже тридцать-сорок лет, научила нас этому. Где сегодня DEC? Где сегодня Wang? Где Data General? Этого не произойдет. Где-то лучше использовать сервер Novell, где-то – сервер Red Hat, где-то, возможно, все еще подходит Sun, а где-то – Windows. Каждый потребитель выбирает то, что нужно его бизнесу. И вы не можете попросить их что-то порушить.

О СТРАТЕГИИ XANDROS

«Мы не можем пойти и заняться тем же, чем занимаются все подряд.»

LXF: По-вашему, Xandros отличается от Novell или Red Hat философией?

AT: Я думаю, то ценное, что предлагаем мы, разработчики дистрибутивов, заключается в нашем вкладе в сообщество – улучшении продукта, или участии в его разработке – и в обеспечении добавленной стоимости [эх] поверх всех усилий сообщества, чтобы этот продукт, эта платформа быстро прижилась в реальном мире.

Наша точка зрения заключается в признании двух простых истин: а) мир гетерогенен, б) не стоит тратить время на то, чтобы соединять или интегрировать что-то – все должно быть в духе «Plug and Play». Вы не можете заявить: «Ну, ладно, берите Linux, но вам понадобятся пятеро инженеров, чтобы его установить, и еще трое, чтобы сообразить, что делать, если завтра система откажет». Если у вас есть администраторы, привыкшие к Windows, вам придется использовать их. Так что наша основная философия – это сосуществование, взаимодействие, интеграция и использование имеющихся умений и навыков.

Возьмем рынок малого и среднего бизнеса (SMB). Это в первую очередь рынок *Windows Server*. Почему так? А потому, что большинство поставщиков решений имеет больший навык работы в Windows, чем в Linux. Они не могут позволить себе сертифицироваться у Red Hat или найти инженеров с сертификатом Red Hat, прежде чем продать сервер за $500. Они не могут прийти к своему клиенту и сказать: «Кстати, вам будет нужен навык работы в Linux или администраторы с опытом работы в Linux». Это сокращает долю Linux на данном рынке. Вы можете использовать Linux в центре обработки данных в Goldman Sachs или в Citigroup, или еще где-нибудь, но там зачастую он просто приходит на смену Unix – и поэтому уже есть необходимые навыки.

На самом деле, если Xandros и нам подобные хотят способствовать развитию Linux, надо не просто улучшать открытый код, а в первую очередь – создать инструменты. Если у меня есть программисты, которые будут делать патчи к ядру, разве это более ценно, чем создание неких инструментов, [таких, чтобы] вдруг весь мир смог использовать Linux?

LXF: Это зависит от того, что делают ваши патчи к ядру!

AT: Я считаю, что мы должны заниматься и тем, и другим. Дело ведь не в том, чтобы просто брать открытый код, упаковывать и отдавать кому-то.

LXF: Сколько у вас разработчиков?

AT: По-моему, у нас почти 60 человек в США, несколько в Европе, и около 15 в Индии.

LXF: И Xandros явно больше не является просто компанией по разработке настольных дистрибутивов?

AT: Нет, нам удалось усовершенствовать то, чего мы добились в этой области и создать серверный продукт для рынка SMB [Xandros Server], с которым может работать администратор Windows. Например, если у вас есть небольшая компания, и вам надо изменить один сервис на своем сервере – допустим, брандмауэр – вам (в отличие от инженера) незачем знать, что если вы внесете это изменение, то надо поправить что-то и в этом месте, и в том месте, и еще в десяти местах, чтобы все заработало. А если вы – инженер и сделали изменение в одном месте, вдруг вам звонит жена и говорит: «У нас ребенок заболел, приезжай домой»? Надо срочно встать и ехать, а пока вы вносили изменение здесь, все в целом вышло из строя.

LXF: Вы считаете, что основной барьер, который мешает людям принять Linux – их страх, что у них нет нужных знаний и умений?

AT: Конечно. Но я думаю, что для рабочих столов тот факт, что некоторые упрощают использование своего продукта, делают его более «Plug and Play», означает, что пользователю не придется учиться чему-то новому – больше нет командной строки и т.п. Это также означает, что создатели приложений могут сказать: «Я могу выпустить свой продукт под Linux, потому что сейчас стали покупать больше машин с Linux», что само по себе в какой-то степени подталкивает производителей ПК к тому, чтобы поразмыслить над этим. Но на самом деле, это Уловка-22 [название романа Джозефа Хеллера, используемое для обозначения ситуации, не имеющей решения, – прим. пер.], проблема курицы и яйца.

На предприятии, или в сфере SMB, я считаю, переходу будет способствовать создание инструментов, которые позволят снизить стоимость управления. Я не хочу запоминать пять вещей, без которых ничего не заработает. Нужно взять кусок там, кусок сям? Нет уж, упростите это для меня. Предоставьте мне это в интуитивно понятном виде, простом и ориентированном на бизнес. Я не желаю знать, что там внутри.

LXF: Вам просто нужна маленькая симпатичная панель управления, чтобы вы могли видеть, например, Email – да, мне это нужно; Sharing – да, частично нам это понадобится; и кнопку «Все Исправить», на случай неполадок.

AT: Верно. Вы хотите сказать: «Запихайте все в черный ящик и позвольте мне заняться управлением в целом, невзирая на то, на каком сервере это находится». Это и есть реальный мир. Вот этого вы и хотите, будучи пользователем, в противоположность мне, который говорит вам, что для управления разными серверами надо помнить об определенных инструментах. Вот в чем заключается наша идея: есть одна-единственная панелька. Этот «черный ящик» превращается в движок, и затем может работать на Novell, Windows, Linux или даже в браузере, и соединяться с вашими инструментами управления [Типальдос подразумевает *Xandros BridgeWays*, кросс-платформенный набор инструментов системного администратора].

LXF: Могу ли я задать вам один вопрос об этом? Что произойдет, если на ваш сервер Red Hat установить исправление безопасности, которое изменит работу *BridgeWays*? Как тогда оно будет работать?



AT: Ну, как бы вы поступили на данный момент? Вы бы отправились на сайт Red Hat Network, взяли оттуда заплатку и распространили ее. И вам бы следовало помнить, что в другом отделе у вас стоит Novell, а в третьем – Xandros, а еще некоторые работают в Windows. Почему бы не обладать одной консолью – вроде капитанского мостика – и говорить с него: «Отправляйтесь на Red Hat, возьмите патч и отправьте его на серверы Red Hat; затем идите на Novell, получите патч и отправьте его на серверы Novell; и то же самое для Windows»? Иными словами – заставьте работать ПО, не занимайтесь этим вручную.

LXF: Да, я понимаю.

AT: Мы полагаем, что наш вклад в развитие Linux таков: люди хотят, чтобы мы облегчили их жизнь. Они хотят, чтобы у них был выбор, но только не ценой осложнений. Если им сказать: «Вот, у вас есть выбор, но теперь вам придется освоить еще одно руководство», они вам ответят: «Знаете что? Я лучше останусь при своем. Лучше я обновлю мой Windows PC до Vista вместо того, чтобы учиться чему-то новому».

LXF: По вашему мнению, в Linux на данный момент слишком большой выбор из-за большого количества дистрибутивов? Не стал бы мир проще, если бы имелись только SUSE, Red Hat…

AT: Думаю, эволюция нас научила тому, что в некий момент происходит уничтожение и консолидация. Сейчас IBM и HP и другие проталкивают исключительно Red Hat и Novell. Очень трудно становиться третьим выбором.

LXF: Однако же HP сейчас поддерживает еще и Debian.

AT: Правильно. Я думаю, что, если у вас на самом деле появится то, что я описал раньше [философия laissez-faire Типальдоса («пусть делают как знают»), – прим. пер.], вы сможете позволить себе такой выбор. Круг сужается потому, что никто не может позволить себе слишком широкий выбор. Сколько раз вы приходили в магазин и говорили: «Жаль, что у вас не 2 продукта, а 22? Надо узнавать обо всех функциях и свойствах, сравнивать… просто скажите мне, что делать». Так что я думаю, выбор будет ограничен, если на самом деле ценой, которую вам придется заплатить за этот выбор, будет сложность.

Наш подход, как мы полагаем, заключается в поддержке большего выбора. Но, в конце концов, для многих компаний станет трудно выжить, если они не смогут поддерживать солидных исследований и разработок. Я хочу сказать, что никто не сравнится своим богатством с Red Hat или Novell. У большинства дистрибутивов нет солидных ресурсов. В нашем случае, на исследования и разработку ушло $60 млн. Но многие ли могут себе позволить потратить 60 млн.? Чтобы и дальше наращивать свою ценность, требуется много денег, и вы не можете позволить себе быть, как все. Я думаю, это все приведет ко все большей консолидации и уменьшению выбора. По-моему, это также приведет ко все большей концентрации на повышении качества и дополнительных возможностях.

LXF: Что вы думаете об Ubuntu сейчас, когда у них появилась LTS-версия с долгосрочной поддержкой, которая в большей степени ориентирована на рынок серверов, чем рабочих столов?

AT: Ясно, что Ubuntu стал феноменом, и это хорошо для нашей индустрии. Но в конечном итоге все опять сводится к основному вопросу: если вы – глава отдела продаж в крупном инвестиционном банке в Лондоне или Нью-Йорке, станете ли вы рисковать своей работой, или своими клиентами, и строить бизнес только на Open Source потому, что вашим технарям больше нравится Ubuntu, чем OpenSUSE или Red Hat? Я думаю, что в конце концов все хотят стабильности, им нужна добавленная ценность на OSS, поэтому...

LXF: Однако же, в какой-то степени эта проблема изначально присутствовала и у SUSE и у Red Hat. Если бы вы работали в крупном банке, стали бы вы рисковать своей работой, выбирая какой-то там Linux со всякой ерундой, в то время, как все вокруг…

AT: Тут есть разница. Red Hat были первыми на рынке, у них был миллиард долларов в банке, и они могли использовать эти свои ресурсы. За всем этим стояла фирма. Если вы возглавляли некую инфраструктуру в Deutsche Bank, вы могли позвонить Мэтту Зулику [Matt Szulik, глава Red Hat], и если ему требовалось прислать вам целую армию инженеров, он их присылал. А теперь наоборот, представьте, что некто сообщает вам, что скачал OpenSUSE, Xandros или Ubuntu, и теперь вся ваша важнейшая система торговли теперь работает на нём. Вы говорите: «Минуточку,



а если ты завтра уйдешь, что тогда будет? У меня останутся те, кто работает в Windows – к кому мне обращаться?»

LXF: Ubuntu определенно много сделал для рабочих столов.

AT: Конечно. Debian по-прежнему остается самой успешной платформой, и, возможно, Ubuntu подталкивает Debian к тому, чтобы заявить: «Ладно, пусть будут более регулярные обновления, и т.п.» Но для меня… Если я скажу вам «Cisco», что вы подумаете? Cisco – это фирма, которая была создана поверх открытой системы – TCP/IP. Но вы думаете о дополнительных функциях и качестве. Или возьмите Oracle. Это фирма, созданная благодаря документу о реляционных базах данных, опубликованному IBM 30 лет назад. Но ценность Oracle не в том, сколько изменений они внесли в фундаментальную концепцию реляционных БД. Вот почему, в конечном итоге, наш подход заключается в том, чтобы содействовать переходу на Linux посредством дополнительных возможностей и качества, которые упрощают внедрение. Мы не можем пойти и заняться тем же, что делают все подряд. В Red Hat делали это, в Novell делали это, Ubuntu делает это сейчас – «Вот вам Open Source, и мы даем вам договор на поддержку». И у вас есть Oracle.

LXF: Да, мне очень интересно, как сыграет Unbreakable Linux от Oracle. Скажите мне, как вы думаете, где будет Xandros в это же время в следующем году?

AT: Мы надеемся увидеть плоды своих воззрений: увеличить долю Linux в SMB-сфере и затем – улучшить интероперабельность в корпоративной среде. На данный момент мы работаем в Европе, и мы расширяемся: мы уже работаем в Латинской Америке. Мы хотим, чтобы на нас смотрели не просто как на фирму, делающую настольные решения, но как на фирму, которая [распространяет] свои достижения на сервера, на уровень управления, на уровень приложений. Точно также Microsoft начинала с рабочих столов, а превратилась в компанию серверов и приложений. Я хочу, чтобы о нас думали так: «Red Hat, Novell… Xandros». Это основная проблема позиционирования, проблема имиджа, но во-вторых, мы хотим, чтобы люди смогли увидеть, как организации работают с нашими продуктами. И я верю, что если так и будет, мы внесем свою лепту, выравнивая поле рынка таким образом, чтобы люди не боялись, что у них появятся сложности, если они перейдут на Linux. Так что мы несем Linux туда, где его раньше не было. Это как *Starship Enterprise*…

Что за штука... Е?

Это действительно язык программирования нового образца, или просто продолжение алфавитной линии C и D? **Пол Хадсон** все объяснит…

Значит, вы не шутили, говоря об этом в прошлом номере?

Ничуть. В LXF93 мы рассматривали D, созданный на замену C и C++. Теперь мы говорим о Е.

Язык на замену D? А я-то думал, что D – это новый язык!

Вы правы, D и в самом деле новый язык – но Е совсем не предназначен для его замены. D способен заменить C потому, что похож на него лингвистически и тоже компилируется в высокоэффективный машинный код. Язык Е, напротив, интерпретируемый (во всяком случае, пока), вдобавок он совершенно не похож на C по стилю программирования – фактически, это странная смесь Java и Python. Предупреждая ваши вопросы: нет, тот Е, о котором мы с вами говорим, не имеет ничего общего с языком программирования Е, некогда бывшим основным на Amiga. На другой напрашивающийся вопрос: да, нам бы тоже хотелось, чтобы создатели языков программирования были чуть более изобретательны в выборе названий!

> «Е имеет все средства управления, обычные для других языков, но некоторые операторы доработаны для удобства восприятия.»

Хорошо, хорошо. Но чем же Е отличается от остальных языков программирования?

Е предназначен для безопасного распределенного программирования, и это его козырь: единственная программа может работать на многих машинах.

А разве этого не было раньше? Пока мы тут разговариваем, мой SETI@Home занят поиском внеземных цивилизаций…

На языках Java и C++ распределенное программирование реализовывалось индивидуально каждым программистом. В Е распределенное программирование – это стандарт, потому что объекты…

Опять эти объекты! Объясните мне, наконец, почему все современные языки программирования – объектно-ориентированные?

Хорошо, давайте остановимся на этом. Да, Е – объектно-ориентированный язык. Фактически, он более объектно-ориентированный, чем другие языки программирования, потому что все в Е подчинено ООП.

Сейчас завою…

Да ладно, это я так, про себя. Итак, Е предназначен для ООП, и не напрасно: именно объекты придают ему вычислительную мощь. Например, объект 1 работает на компьютере 1, а объект 2 на компьютере 2. Если объект 1 вызывает метод объекта 2, то Е автоматически формирует запрос, отправляет сообщение по сети и возвращает результат обратно. Е-программист даже не обязан знать, где находятся объекты – на его машине или на чужих: Е берет все это на себя.

[фырканье] Хм, еще того не легче: а как же безопасность?

Создатели Е позаботились об этом заранее – все сетевые сообщения шифруются и недоступны посторонним.

Ладно, объекты, может, и не столь ужасны. Зато я уверен, что обмен сетевыми сообщениями абсолютно всегда приводит к взаимным блокировкам и зависанию.

На самом деле это не так – у Е нет таких проблем.

Ой ли? А вы уверены, что мы говорим об одном и том же?

Конечно, уверен: взаимная блокировка возникает, скажем, тогда, когда у меня есть LXF90, а мне нужен LXF91; у вас же, наоборот, есть LXF91, а вам нужен LXF90. Никто из нас не хочет отдавать свой номер, не получив чужого – вот вам и тупик. Ну что, на одном языке мы с вами говорим?

Вроде да. И как Е это удается?

Магия! Точнее, колдовство под названием «обещания», позволяющее мне пообещать отдать вам свой LXF90, если вы отдадите мне LXF91. Как только вы это делаете, я тут же отдаю вам LXF90, и все довольны. Теперь вообразите следующий код: someobject->doStuff(). В большинстве языков программирования (включая Е) программа не будет продолжена до тех пор, пока someobject не выполнит свой метод doStuff(). В Е это называется прямым вызовом функции. Но в Е есть и другой способ, под названием «отложенный вызов», то есть вызов, не блокирующий работу программы.

LINUX FORMAT на DVD
» E 0.9.1b

Ого, а это еще что?

Давайте вернемся немного назад. Так вот, someobject->doStuff() исполняется немедленно, вынуждая остальную часть программы ждать. Отложенные вызовы, которые выглядят примерно так: someobject<-doStuff(), не требуют мгновенного исполнения. Фактически, вы говорите: «someobject, когда будет возможность, пожалуйста, выполните doStuff()». В результате программа, сделавшая этот вызов, беспрепятственно продолжает свою работу.

Ага! Проблема налицо: что произойдет, когда doStuff() вернет нужное значение?

Извиняюсь, создатели Е тоже об этом подумали! Если вы попытаетесь воспользоваться результатом, возвращенным doStuff(), во время выполнения непрямого запроса, то фактически получите обещание, что нужное значение будет (при необходимости) вычислено. Обещание не будет выполнено до тех пор, пока doStuff() не будет реально вызвана и не возвратит нужное значение, а до тех пор у вас будет лишь его «заготовка». Но зато обещание гарантирует, что вы получите нужное значение в будущем, а пока работа программы будет продолжена так, словно оно уже известно.

Как-то в голове не укладывается.

Бесспорно, уяснить это непросто. Вспомните пример с журналом – мы просим LXF91, но, поскольку некий человек не хочет отдавать его сразу, вместо журнала мы получаем обещание. Для нас оно в перспективе равносильно LXF91, и мы смело отдаем этому человеку свой LXF90. Как только это произойдет, обещанное воплотится в реальный LXF91.

Ладно, поверю вам на слово, что все это работает. Расскажите еще что-нибудь про Е – только попроще, пожалуйста!



Как вам вот это: в нем остались все средства управления, к которым вы привыкли в других языках программирования, например if, try, catch, finally, while и for. Но некоторые операторы для удобства восприятия несколько доработаны. Например, чтобы избежать путаницы между = и = =, E перенял подход Pascal и использует := для присваивания, а = = для сравнения (сам по себе знак = больше не применяется). E, в основном, обходится без переменных с типом (string, integer и т.п.), так как подразумевается автоматическое преобразование типов. Например, если добавить к строке число, то E конвертирует число в отдельную строку, а затем объединит обе строки.

А разве языки без типов не усложняют программирование?

Это распространенная точка зрения, и, во избежание проблем, E снабжает программиста «предохранителями» типов, напоминающими признаки классов в PHP. Например, если необходимо, чтобы функция обрабатывала только числа, можно указать это прямо: все входящие данные будут конвертироваться в этот формат. Если конверсия невыполнима, E выбрасывает исключение.

Если кто-нибудь – только не я – захочет попробовать E, как это можно сделать?

Лучшая версия E построена на основе Java, что имеет два основных достоинства. Во-первых, она может работать везде, где есть Java, то есть на Linux, Windows, OS X, мобильных телефонах – да где угодно. Во-вторых, вы получаете функциональность Java простым импортом библиотек. Это означает, что в E можно пользоваться Swing, если захочется – до тех пор, пока вы не поймете, что используемые вами классы Java ничего не получают от распределенной мощи E.

Но я ненавижу Java!

Не беспокойтесь: вам и вашим друзьям – Java-ненавистникам будет приятно узнать, что существует версия E на Common Lisp, которая действует примерно так же – минус поддержка Java, конечно!

Обе версии свободны?

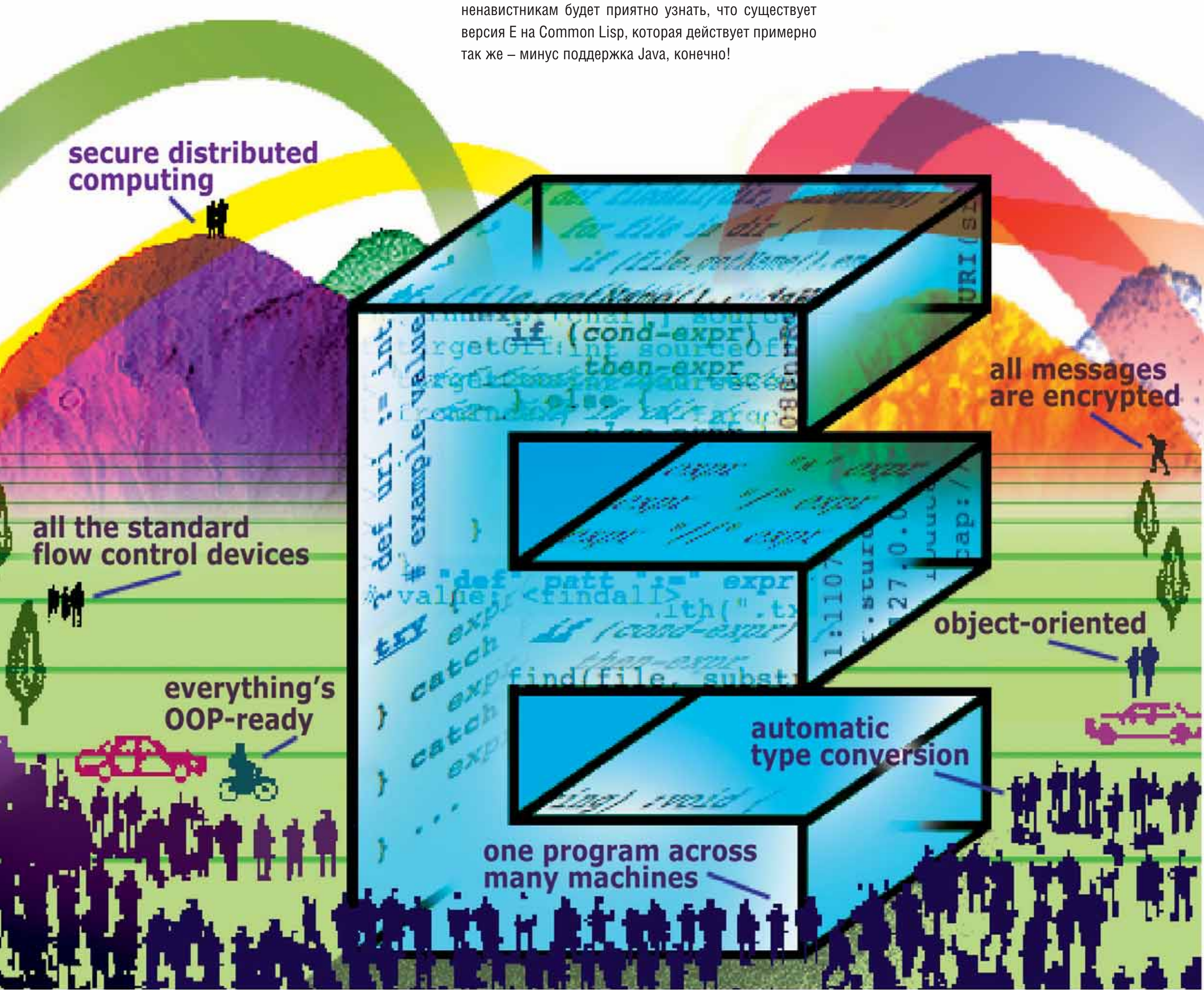
Конечно – под Mozilla Public License.

Круто. Пару слов напоследок?

Если вы хотите узнать об этой теме подробнее, можете посетить такие сайты: www.erights.org – официальная страница, www.skyhunter.com/marcs/ewalnut.html – бесплатный онлайн-учебник по E, и www.combex.com/tech/edesk.htm – домашняя страница CapDesk, рабочей среды, полностью написанной на E, чтобы доказать высокий уровень его безопасности.

Спасибо. Итак, через месяц говорим об F?

Поживем – увидим… LXF



КОНКУРС дистрибутивов

Если заглянуть за номера версий, то каковы реальные различия между лучшими на сегодняшний день дистрибутивами? Пол Хадсон, Грэм Моррисон и Ник Вейч найдут для вас идеальный Linux...



Linux предлагает целый калашный ряд дистрибутивов, подходящих к любому ПК – этот выбор есть часть прелести Open Source. Однако, при таком богатстве (по оценкам – не менее 400 видов) как избежать неправильного выбора?

Фактически дистрибутивы Linux сильно отличаются своим подходом к безопасности, размером, производительностью, совместимостью с аппаратным обеспечением и десятком других ключевых факторов, которые волнуют людей. Тем не менее люди часто выбирают дистрибутив из-за более яркой темы рабочего стола или из-за номера, большего на 0.1 – то есть по совершенно неуместной причине. Нет, мы, конечно, не хотим сказать, что уродливый рабочий стол лучше приятного, но что если у первого чаще выпускаются заплатки по безопасности, он вдвое быстрее загружается, правильно определяет ваше оборудование или имеет гораздо большее сообщество? Существует множество вещей, не учитываемых в номере версии, но они-то и определяют разницу между хорошим дистрибутивом и великолепным дистрибутивом.

На следующих восьми страницах мы сравним восемь лучших дистрибутивов и проверим, как они соотносятся по безопасности, производительности, совместимости, доступности приложений и сообществу, чтобы вы смогли найти идеальный дистрибутив для ваших целей. Конечно, у каждого свои представления о «лучшем дистрибутиве», так что мы выбрали восемь дистрибутивов, самых популярных на момент написания статьи: Debian, Fedora, Mandriva, PCLinuxOS, Sabayon, Slackware, OpenSUSE и Ubuntu. Некоторые остались за бортом: SimplyMepis и Linspire были отброшены из-за близости с Ubuntu; Gentoo – из-за того, что его ответвление, Sabayon, похоже, превзошел его в популярности; а дистрибутивы для предприятий отсеяны, потому что это все-таки иная категория.

Чтобы помочь усмотреть различия между дистрибутивами, мы собрали 16 тестов, охватывающих широкий предметный диапазон. Например, сколько портов открыто по умолчанию? Как быстро запускается *OpenOffice.org*? Велико ли сообщество у дистрибутива? Все это размещено под следующими заголовками: безопасность, совместимость с оборудованием, производительность, сообщество и выбор пакетов, и вы сможете увидеть, как каждый дистрибутив соотносится с остальными и, надеемся, найти то, что вам нужно. Заметим: при любом сравнении дистрибутивов, какому-то придется быть последним. Если таковым оказался ваш любимый дистрибутив (мы не будем делать поблажек), утешьтесь тем, что даже последний из восьми все-таки попал в лучшие 2% дистрибутивов Linux!

Контроль версий

При любом анализе дистрибутива следует обозначить, что именно исследовалось. Мы тестировали: Debian 4.0, Fedora Core 6, Mandriva 2007, PCLinuxOS 2007 TR4, Sabayon 3.3, Slackware 11, SUSE 10.2 и Ubuntu 7.04. Все они устанавливались на одно и то же оборудование, с использованием опций по умолчанию.

Безопасность

Правда ли, что Linux неуязвим? Неправда. Некоторые дистрибутивы даже не имеют брандмауэра...

Прошли те времена, когда Linux был операционной системой типа «поставил-и-забыл», которая – если вы следили за обновлениями – была в основном защищена от негодяев. Вероятно, ключ к защищенности Linux в том, что его традиционно устанавливали пользователи более высокого полета. Менее опытных пользователей отпугивали странные установщики, ад RPM, несовместимость с Windows и нехватка простых в использовании утилит настройки. То есть под Linux в большинстве своем работали люди с техническим опытом, а не те, кто дважды щелкает по файлу **ILOVEYOU.vbs**.

Но ситуация меняется: хотя мы и не собираемся повторять заезженную фразу «этот год будет годом Linux для настольных систем!», несомненно, сейчас пользователи настольных компьютеров переходят на Linux чаще, чем когда-либо. В результате системы безопасности Linux проверяются по полной программе. Хорошо ли защищен Вася Новичков от опасностей со стороны Интернета?

Чтобы определить это, мы учли следующее:

- Как быстро выходят заплатки по безопасности для данного дистрибутива?
- Сколько интернет-портов открыто в стандартной конфигурации?
- Позволяет ли дистрибутив легко настроить брандмауэр, если таковой имеется?
- Какое дополнительное ПО по безопасности входит в дистрибутив?

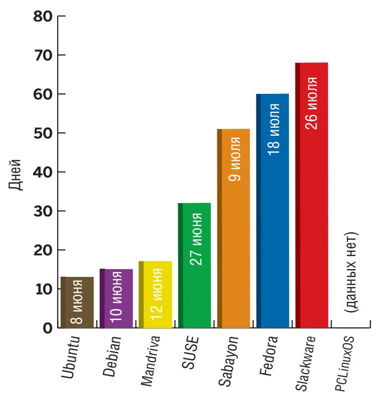
Можно подумать, что результаты будут близки друг к другу, поскольку все дистрибутивы основаны на одних и тех же базовых приложениях. Но это не так: мы проверили одну конкретную уязвимость (CVE-2006-1861) и обнаружили, что разница в выходе заплаток составила шокирующие 55 дней. Самый быстрый, Ubuntu, имел доступную заплатку через 13 дней, а самому медленному, Slackware, потребовалось 68. Хуже того, похоже, что PCLinuxOS даже не выпускает бюллетени по безопасности, то есть вы никогда не узнаете, где слабые места вашей системы. Fedora – единственный поставщик, у которого есть оправдание медленной реакции, поскольку при исправлении этой проблемы команда распознала другую, которую упустили все остальные, и другим поставщикам позже пришлось выпускать заплатку для CVE-2006-3467.

Что касается брандмауэров, мы считаем недопустимым, что многие дистрибутивы поставляются без активизированного брандмауэра. Slackware – наихудший из них, так как прославился не только отсутствием брандмауэра, но и тем, что имеет целых пять портов, открытых всему миру.

> «Команда Fedora распознала проблему, упущенную всеми остальными.»

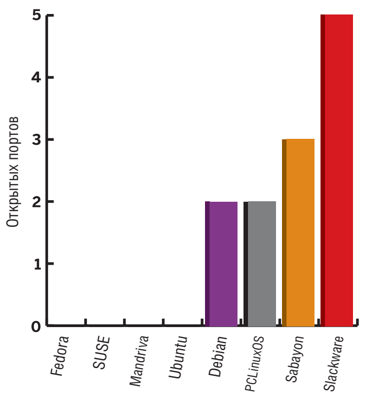
Ubuntu также страдает отсутствием брандмауэра, но у него хотя бы нет открытых портов, то есть риск атаки минимизирован. На другом краю шкалы расположились Fedora, SUSE и Mandriva, при этом дистрибутивы заперты так крепко, что машина при сканировании кажется зависшей. Fedora возглавляет список, поскольку при выключении брандмауэра сервисов остается всего ничего, тогда как SUSE и Mandriva оставляют X11 открытым. Заметим: мы не рекомендуем отключать брандмауэр для ваших личных целей!

Скорость исправления ошибок



Число дней между сообщением об ошибке (19 мая 2006 г.) и выпуском заплатки поставщиком.

Результат сканирования портов



Число портов, открытых в стандартной установке дистрибутива.

Простота настройки

1. SUSE
2. Fedora
3. Mandriva
4. PCLinuxOS
5. Ubuntu
6. Debian
7. Sabayon
8. Slackware

Показывает, легко ли настроить брандмауэр через графическое приложение.

Безопасность: итоги

1. SUSE
2. Mandriva
3. Fedora
4. Ubuntu
5. Debian
6. PCLinuxOS
7. Sabayon
8. Slackware

»



Совместимость с оборудованием

Вопрос на миллион. Какой дистрибутив сработается с вашим оборудованием?

Большинство дистрибутивов Linux лезут из кожи вон, чтобы поддерживать как можно больше устройств, и ситуация сейчас намного лучше, чем раньше. Но, к сожалению, все еще бывает и так, что единственный способ проверить работоспособность чего-нибудь – это его попробовать. И даже если вам удалось заставить ваше устройство работать в вашей системе, нет гарантии, что оно заработает у другого. Природа Linux такова, что все можно заставить работать почти на всем, а число возможных сочетаний оборудования практически бесконечно.

По этой причине мы не могли взять и протестировать каждую мыслимую аппаратную комбинацию. Результаты были бы интересными, но они не сказали бы вам, заработает ли ваш новый USB-что-то-там на самом свежем дистрибутиве. Вместо этого мы сосредоточились на ядре Linux как на сердце каждой системы. Именно в ядре содержится большая часть драйверов, требующихся для работы оборудования. Подавляющая часть драйверов реализуется как модули ядра, при этом каждый модуль обслуживает одно устройство. Подсчет числа модулей, входящих в дистрибутив – хороший способ оценить спектр поддерживаемого оборудования. Аналогично, глубина может оцениваться по версии ядра. Чем новее ядро, тем более вероятно, что оно содержит обновленные драйверы, исправленный код и улучшенную поддержку новых устройств.

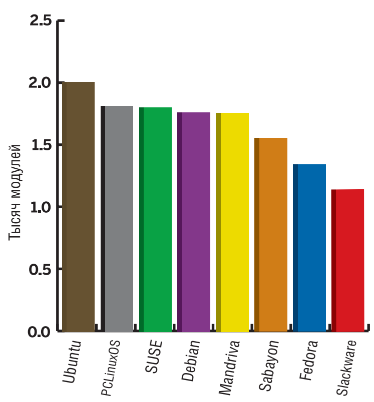
У каждого дистрибутива свой подход к включению драйверов. Например, Fedora всегда была осторожна в этом вопросе, ибо драйверы способны нарушить стабильность. Если устройство работает под этим дистрибутивом, у вас будет меньше проблем, чем в Ubuntu, который шлет все предосторожности лесом и включает поддержку на 50% больше устройств (если сравнивать последние версии). Подход Ubuntu хорош для любителей новизны, но он утрачивает прелесть, если вам требуется, чтобы машина работала 12 месяцев подряд без перезагрузки.

Однако модули ядра – еще не вся сказка. Многие дистрибутивы для предприятий имеют свою собственную базу данных по совместимости, чтобы помочь бизнесу принять обоснованные решения насчет оборудования, перед тем как вставлять установочные диски. Fedora, SUSE и Mandriva публикуют списки официально поддерживаемого оборудования, обязанного работать с данным дистрибутивом. Жаль, что Linux-сертификация оборудования не стала стандартом. Но теперь, когда можно купить компьютеры Dell с предустановленным Ubuntu, существует стандартная аппаратная платформа, на которую можно опереться, и совместимость все время растет. Что нам действительно нужно от производителей оборудования, так это чтобы они обратили внимание на Linux и открыли свою интеллектуальную собственность и аппаратные спецификации для разработчиков, которые желают писать драйверы. Это ведь не только поможет повысить продажи, но и значительно упростит нам жизнь.

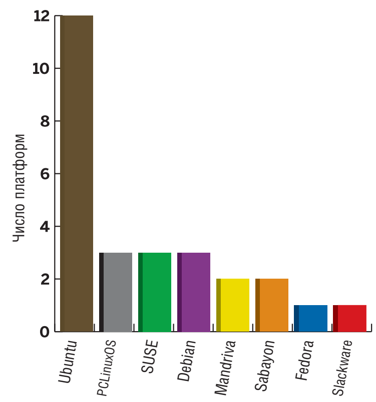


› Mandriva, как всегда, щеголяет классными утилитами настройки оборудования.

Драйверы-модули



Поддержка платформ



Версия ядра

Версия ядра Linux, включенного как стандартное.

1	Ubuntu	2.6.20.15
2	Sabayon	2.6.20.3
3	Debian	2.6.18.4
4	PCLinuxOS	2.6.18.4
5	SUSE	2.6.18.2
6	Fedora	2.6.18.1
7	Mandriva	2.6.17.13
8	Slackware	2.4.33.3

Оборудование: итоги

1. Ubuntu
2. Debian
3. SUSE
4. PCLinuxOS
5. Sabayon
6. Fedora
7. Mandriva
8. Slackware

Текущий итог

1	SUSE	‹›
2	Ubuntu	↑ 2
3	Debian	↑ 2
4	Mandriva	↓ 2
5	Fedora	↓ 2
6	PCLinuxOS	‹›
7	Sabayon	‹›
8	Slackware	‹›

Производительность

Параметры компиляции, версии ядра, конфигурация жесткого диска и прочее – все это влияет на работу дистрибутива.

Почему Red Hat 7.2 спокойно работала при 64 МБ памяти на Pentium первого поколения, а Fedora Core 6 еле ползет на всем, что хуже, чем 1.8-ГГц Pentium 4 с 256 МБ оперативки? Сравнение, правда, не совсем справедливое: хотя они запускают одни и те же основные программы и одинаково хороши для тех же задач, FC6 и другие современные дистрибутивы делают сильный упор на хороший внешний вид и применение последних технологий, экономящих время. *Beagle* – хоть и полезен – означает загрузку Mono для работы с .NET; *GTK 2* – хоть и красив – подразумевает отображение виджетов с использованием Cairo для красивых сглаженных линий. Потом, конечно, есть еще и *Compiz*...

Но каждый дистрибутив работает с этими компонентами по-своему. Некоторые, например, Fedora и SUSE, похоже, включают их все, а затем бросают все силы на борьбу с перерасходом ресурсов. Другие, такие как PCLinuxOS и Slackware, приняли более экономный подход: начальная конфигурация делается легковесной, но позволяет вам добавлять другие функции самостоятельно.

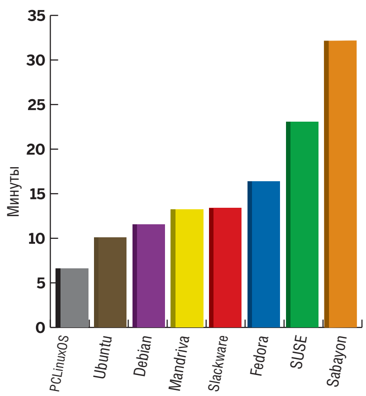
Для проверки производительности мы учли 4 фактора: как долго устанавливается каждый дистрибутив (вплоть до первого приглашения входа в систему), сколько времени занимает загрузка (опять-таки, до появления экрана входа), насколько быстро выполняется холодная загрузка *OpenOffice.org* (то есть непосредственно после перезагрузки), и сколько свободной оперативной памяти имеется в системе сразу после перезагрузки (измерялось с помощью **free -m**).

Мы обнаружили заметные различия в производительности. Три теста на скорость показали, что самые быстрые дистрибутивы работают почти в 2 раза быстрее, чем самые медленные, а самый жадный до памяти дистрибутив съедает ее на 120 МБ больше, чем самый неприхотливый, даже при отсутствии запущенных пользовательских приложений.

Все это показывает, что правильным выбором дистрибутива можно добиться многого, независимо от имеющегося оборудования. Наши тесты проводились на двухядерном Intel Core 2 Duo с тактовой частотой 2 ГГц и 1024 МБ ОЗУ, со свежими версиями каждого дистрибутива. В итоге обнаружено, что PCLinuxOS 2007 лидирует по общей производительности: время установки менее семи минут весьма эффектно, особенно если учесть, что стандартные программы типа *OOo* включены в поставку. PCLinuxOS занял первое место и по количеству свободной оперативной памяти: ему удалось переплюнуть даже супер-умеренный Slackware, что было для нас большим сюрпризом. Будучи ближайшей родней PCLinuxOS, Mandriva также хорошо показала себя, имея наименьшее время загрузки.

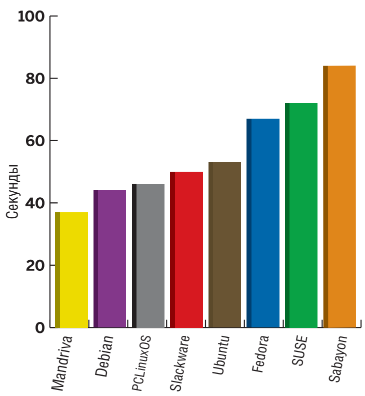
На другом конце шкалы расположились Sabayon и Fedora, с одинаково плохими результатами всех наших тестов. SUSE – почти на дне таблицы – хотя бы быстрее загружает раз запущенные приложения, что извиняет его за тягучую установку и полусонную загрузку. А вот по части потребления оперативной памяти Fedora просто распоясалась: чтобы просто добраться до рабочего стола Gnome, потребовалось 202 МБ из 1 ГБ ОЗУ, и это объясняет, почему 512 МБ – трезвая рекомендация для Core 6.

Время установки



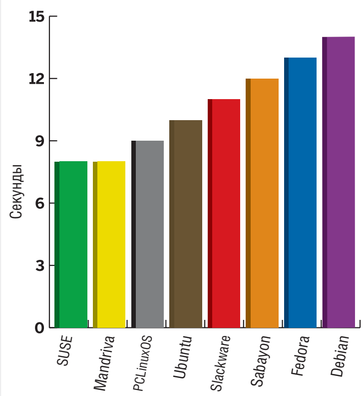
Время, требуемое для установки по умолчанию с использованием рекомендуемых опций.

Время загрузки



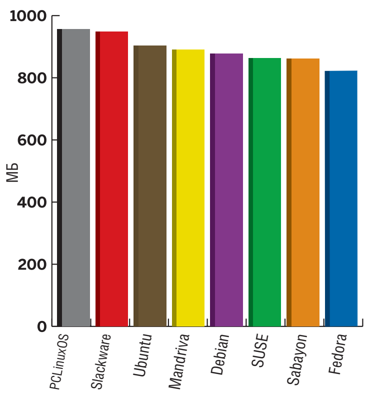
Время, проходящее от приглашения Grub до графического экрана входа в систему.

Запуск OOo



Время первого запуска OpenOffice.org («холодный старт»).

Свободная память



Сколько памяти, в МБ, остается свободной после запуска.

Производительность: итоги

1. PCLinuxOS
2. Mandriva
3. Ubuntu
4. Slackware
5. Debian
6. SUSE
7. Fedora
8. Sabayon

Текущий итог

1. Ubuntu ↑ 1
2. SUSE ↓ 1
3. Mandriva ↑ 1
4. PCLinuxOS ↑ 2
5. Debian ↓ 2
6. Fedora ↓ 1
7. Sabayon ‹›
8. Slackware ‹›

»

Сообщество

Одному всего не сделать – лучшие дистрибутивы имеют больше пользователей, больше поддержки и всякого такого.

Большое значение в Linux придается понятию сообщества: кооперация и прочие прелести. Но в нашем тесте мы все это проигнорируем. Мы просто хотим узнать, что сообщество может сделать для вас. Процветание сообщества означает, что вашей ОС уделяется много внимания со стороны поставщиков и web-провайдеров; сообщество может служить источником советов и подмоги в беде и снабжать вас свежими интересностями из мира Open Source.

Расчет силы сообщества, с любой степенью точности, – сложная задача, поскольку способ измерения отсутствует. Для наших целей мы рискнули выдумывать формулу, способную дать ответ: **Сообщественность = Популярность * Поддержка * Активность**.

Примерно так. Теперь, зная формулу, остается лишь определить значения соответствующих величин. Чуете, в чем проблема? Но кое-что померить можно. Допустим, популярность: Distrowatch.com на своем сайте подсчитывает количество посещений домашних страниц проектов. Есть также статистика на сайте www.linuxformat.co.uk и данные от проекта LinuxCounter. Годится не вся статистика: например, Linux Counter (http://i18n.counter.li.org) не имеет данных о двух членах нашего списка. Так что мы привели все значения к диапазону от 0 до 100 и усреднили по числу источников данных для тех дистрибутивов, у которых более одного источника. Фактически это означает, что часть чисел «надежнее» других, но также и то, что для большинства проектов ни один источник данных не является слишком определяющим.

Самый популярный дистрибутив определен; а как тогда измерить поддержку? Она имеет две формы: взаимопомощь, которую мы измерили по анализу трафика на разных форумах, и официальная поддержка – время, затраченное на выпуск обновлений и прочего. Наконец, мы учли возможность платной поддержки. Существуют консультанты, предлагающие поддержку практически любой версии Linux, но мы должны отдать должное дистрибутивам, которые обеспечивают свою собственную платную службу поддержки. Неудивительно, что дистрибутивы с «большим именем» с корпоративной поддержкой хорошо котируются в этой категории. Например, Mandriva имеет очень хорошо развитое «клубное» сообщество (требующее, однако, подписки); у Fedora есть обширная Вики.

По части активности, мы измерили такие вещи, как объем сопровождаемых пакетов (то есть contrib'ов или RPM'ов от третьих лиц), доступность пакетов для ключевых приложений и прочее.

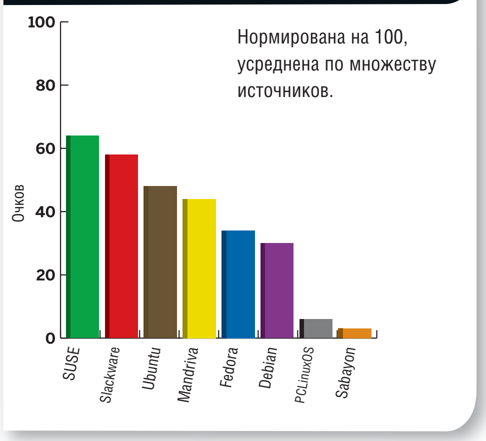
Результаты довольно интересны по нескольким причинам. Наш способ подсчета очков не дает преимущества более новым дистрибутивам (таким, как Ubuntu и PCLinuxOS), которые имеют больше текущих откликов, но которым недостает предыстории таких старых проектов, как Slackware и Debian. В общем, в этой категории он не является строго научным, но, вероятно, настолько близок к нему, насколько возможно.

> «Это не строгая наука, но она близка к правде.»

Текущий итог



Поддержка



Активность



Сообщество: итоги

1. Ubuntu
2. SUSE
3. Debian
4. Fedora
5. Mandriva
6. Slackware
7. PCLinuxOS
8. Sabayon

Текущий итог

1. Ubuntu ‹ ›
2. SUSE ‹ ›
3. Debian ↑ 2
4. Mandriva ↓ 1
5. PCLinuxOS ↓ 1
6. Fedora ‹ ›
7. Slackware ‹ ›
8. Sabayon ↓ 1

Выбор приложений

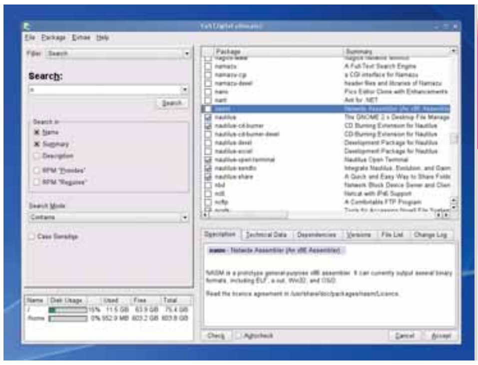
Компиляция собственного приложения – штука медленная и подверженная ошибкам, так что выбор дистрибутива, способного делать это за вас – важен...

Эта категория довольно сложна для оценки, поскольку одни люди предпочитают «тощие-и-легкие» дистрибутивы, где можно установить приложения по мере надобности, другим же нравятся системы, напичканные возможностями, чтобы менеджер пакетов и запускать не пришлось. Наш взгляд на это таков, что дисковое пространство сейчас фактически бесплатно – если у вас есть жесткий диск по крайней мере на 40 ГБ (это на данный момент доступно практически всем), вы сможете установить сколь душе угодно приложений, не задумываясь о расходе места, так что чем больше пакетов – тем лучше. Если же вы намерены использовать Linux на старой машине или хотите хранить множество больших файлов (видео, музыки или фото), тогда о дисковом пространстве стоит побеспокоиться.

Среди тестовых дистрибутивов при стандартной установке меньше всех места на диске занял PCLinuxOS, чей совокупный программный вес составляет чуть меньше 2 ГБ. На другом конце шкалы – Sabayon с почти 9 ГБ приложений, устанавливаемых по умолчанию (чем и объясняется длительность инсталляции этого дистрибутива). Конечно, это стандартная установка, во время которой мы щелкали OK или Next, всякий раз соглашаясь на рекомендуемую опцию.

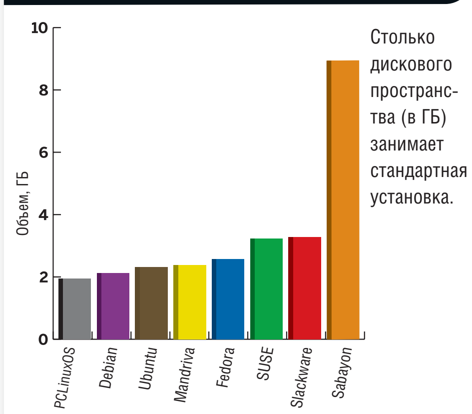
Некоторым важен размер собственно установки, но для нас главным было узнать, много ли приложений можно достать, пользуясь менеджером пакетов дистрибутива. И снова налицо широкий разброс: Slackware сполз на последнее место с жалкими 544 доступными пакетами (это число можно повысить до 1 351, если включить неофициальные пакеты с www.linuxpackages.net), а Debian научил всех жить, обеспечив по умолчанию 18 059 пакетов. Ubuntu имеет такой же охват, но официально поддерживается только его треть. С точки зрения подбора пакетов единственный сюрприз преподнес Slackware, использующий злополучный пакет *KOffice* вместо *OpenOffice.org* – такой выбор только в нем и сделан.

Наконец, многих интересует еще один фактор: как часто они смогут получать новые приложения; и для его оценки мы подсчитали число выпусков каждого дистрибутива (стабильных и не очень) с 1 января 2006 г. Debian ожидаемо оказался в конце этого списка, а победителем, к изумлению, стал SUSE – во многом благодаря его новой политике присвоения двузначных номеров релиз-кандидатам, не говоря уже об альфа- и бета-версиях. Если вы фанат инноваций, этот супербыстрый цикл обновлений идеально подойдет вам. Всем остальным он просто поможет убедиться, что перед финальной версией исправлено столько ошибок, сколько возможно.



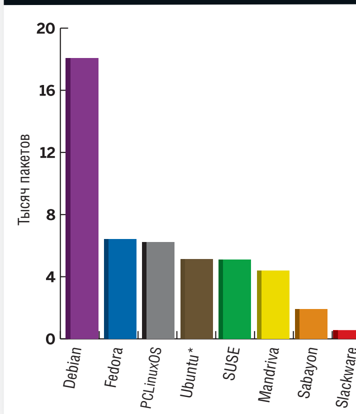
› **Можете не любить *Yast*, но его утилита управления пакетами дает кучу информации по программам, доступным в SUSE.**

Размер установки по умолчанию



Столько дискового пространства (в ГБ) занимает стандартная установка.

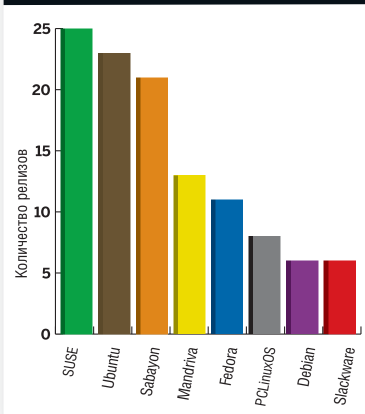
Доступные пакеты



Столько пакетов программ доступно из коробки.

* Указаны только официально поддерживаемые. Дополнительные пакеты доступны в репозиториях Debian.

Частота обновлений



Столько выпусков дистрибутива сделано с 1 января 2006.



Приложения: итоги

1 Ubuntu
2 PCLinuxOS
2 Debian
4 Fedora
4 SUSE
6 Mandriva
7 Sabayon
8 Slackware

Текущий итог

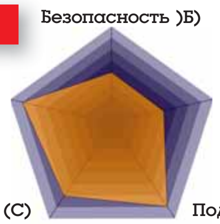
1 Ubuntu ‹›
2 SUSE ‹›
3 Debian ‹›
4 PCLinuxOS ↑1
5 Mandriva ↓1
6 Fedora ‹›
7 Slackware ‹›
8 Sabayon ‹›

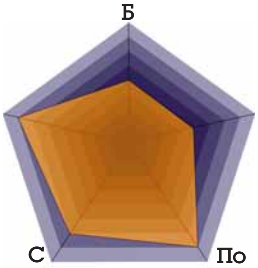
»

Сводная статистика

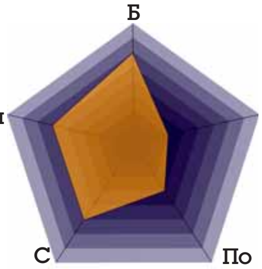
Сильные и слабые стороны каждого дистрибутива и расширенная таблица характеристик, чтобы дать пищу вашему уму...

Легенда:

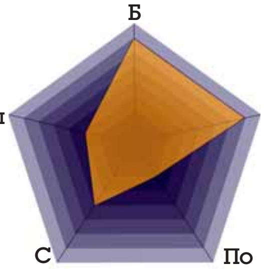




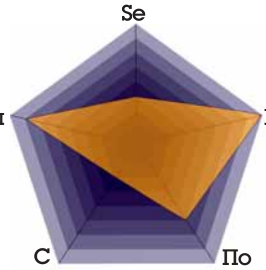
Debian



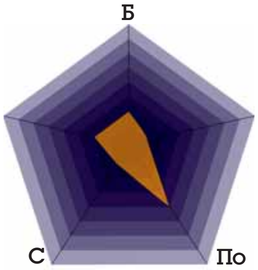
Fedora



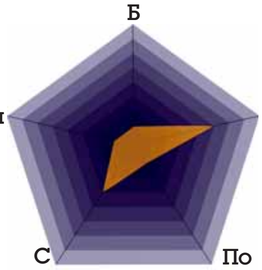
Mandriva



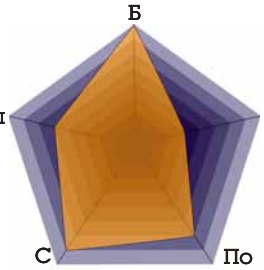
PCLinuxOS



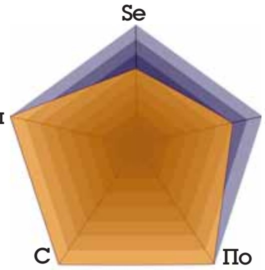
Sabayon



Slackware



SUSE



Ubuntu

Важнейшая статистика для восьми тестированных дистрибутивов

	Debian	Fedora	Mandriva	PCLinuxOS	Sabayon	Slackware	SUSE	Ubuntu
Версия	4	Core 6	2007 Spring	2007 TR4	3.3	11	10.2	7.04
Дата выхода	8/4/07	24/10/06	10/4/07	26/4/07	16/3/07	2/10/06	8/12/06	13/4/07
Сайт	www.debian.org	fedoraproject.org	www.mandriva.com	http://www.pclinuxos.com/	www.sabayonlinux.org	www.slackware.com	www.novell.com/linux/	www.ubuntu.com/
Стандартный рабочий стол	Gnome	Gnome	KDE	KDE	KDE	KDE	Gnome/KDE	Gnome
Производных	124	51	12	0	0	46	4	24
Архитектур	12	3	2	1	2	1	3	3
Управление пакетами	apt	Yum/RPM	urpmi/RPM	apt	Portage	Tarball	Yast/RPM	Synaptic/apt
Размер (ГБ)	2.12	2.58	2.39	1.95	8.94	3.28	3.23	2.31
Пакетов (устанавливаемых)	18,059	6,427	4,392	6,237	1,908	544	5,108	5,132*
Ядро	2.6.18.4	2.6.18.1	2.6.17.13	2.6.18.4	2.6.20.3	2.4.33.3	2.6.18.2	2.6.20.15
X.org	7.1	7.1	7.2	7.1	7.2	6.9.0	7.2	7.2
GTK	2.8.20	2.10.4	2.10.11	2.10.6	2.10.7	2.8.20	2.10.6	2.10.11
KDE	3.5.5	3.5.4	3.5.6	3.5.6	3.5.6	3.5.4	3.5.5	3.5.6

*Указаны только официально поддерживаемые. Дополнительные пакеты доступны в репозиториях Debian..

Итоги

Один дистрибутив явно правит, но кое-кто дышит ему в спину...

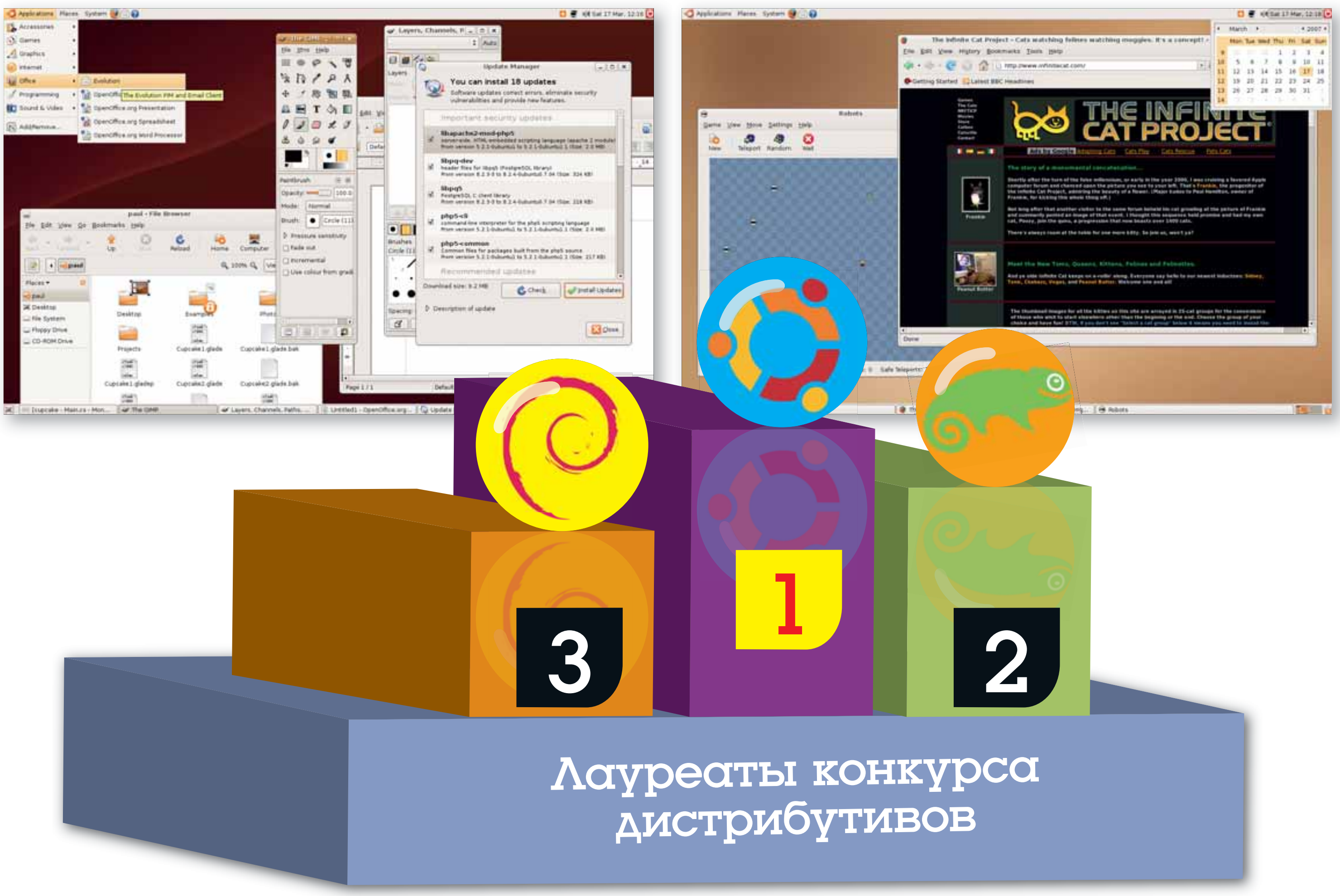
Запускайте ваш почтовый клиент и готовьте убийственные тирады: у нас есть победитель, и имя ему – Ubuntu. Но, надеемся, вы поняли смысл наших диаграмм и таблиц – и то, как мы пришли к этому заключению: Ubuntu не только лидировал в трех тестах из пяти, но и попал в верхнюю часть двух оставшихся таблиц, что и гарантировало ему победу. Бедняга Sabayon, с другой стороны, стал последним или почти последним в нескольких группах – это показывает, что следует приложить больше усилий, чтобы помочь ему догнать самые лучшие дистрибутивы, представленные здесь.

На диаграммах слева хорошо видно, что у каждого дистрибутива есть по крайней мере одна слабая сторона – даже Ubuntu уступил Fedora и SUSE в безопасности, и это позор, потому что пользователям Ubuntu, безусловно, нужен хороший брандмауэр. У SUSE главная проблема – производительность: следует поработать, чтобы уменьшить потребление памяти. Если бы SUSE не ухитрялся выпускать два дистрибутива каждые три месяца, он занял бы более низкое место в программной категории и, несомненно, уступил бы в финальной таблице второе место Debian.

Конечно, если кто-нибудь другой задался бы целью оценить эти дистрибутивы, он, наверное, выбрал бы иные тесты и пришел бы к немного другим итогам. Мы же представили здесь данные, которые могут помочь вам принять свое собственное решение. Надеемся, вы его уже сделали. Что вам нравится в вашем дистрибутиве больше всего? Не убедили ли мы вас подумать о замене? И что следовало бы учесть, а мы этого не сделали? Присоединяйтесь к нашей дискуссии на www.linuxformat.co.uk/distrochat и дайте нам знать! LXF

Финальный итог

1. Ubuntu
2. SUSE
3. Debian
4. PCLinuxOS
5. Mandriva
6. Fedora
7. Slackware
8. Sabayon





Neil Bartlett

Windows в Wine

Никак не можете расстаться с любимым Windows-приложением? Вам поможет *Wine*. О нем-то и расскажет **Майк Сондерс** в первой из двух статей о достижении кросс-платформенной нирваны.

При всех достоинствах Linux и всем многообразии создаваемых для него программ приходится признать, что для разработки ПО платформа Windows остается более предпочтительной. И хотя разработчики Open Source трудятся не покладая рук, чтобы предоставить нам свободную альтернативу популярным программам Windows (в виде *Firefox*, *OpenOffice.org* и *Scribus*), в некоторых случаях полноценного эквивалента просто нет. Linux-версий нет как для крупных программ (*Photoshop* и *Microsoft Office*), так и для тысяч мелких.

Впрочем, выход есть. Это *Wine*. Сегодня речь пойдет не о напитках [Wine (*англ.*) – «вино», – прим. перев.], хотя после бутылки-другой, принятой за один вечер, тоже может показаться, что вы играете в *Quake 8* на Atari ST [популярная в 1985–1990 годах модель настольного компьютера с процессором Motorola, – прим. перев.] в своем домике на Венере. Мы поговорим об одной программе, с помощью которой другие программы, написанные для Windows, можно запускать в Linux.

В традиционном для Unix стиле аббревиатура *Wine* расшифровывается рекурсивно: «*Wine* – Это Не Эмулятор» ('Wine Is Not an Emulator'). Но разве *Wine* не эмулирует Windows?

Нет. На самом деле, эмуляторы воссоздают работу устройства или программы полностью, а *Wine* лишь предоставляет «слой совместимости» между Linux и программами Windows. Когда приложение Windows говорит: «Я хочу открыть файл», *Wine* перехватывает это сообщение и преобразует его в системный вызов, понятный ядру Linux. Это означает, что для запуска программ Windows вам не потребуется ни установленной Windows, ни виртуальной машины типа *VMware* или *Qemu*.

Хотя *Wine* разрабатывается с 1993 года (www.winehq.org), его состояние далеко от совершенства: некоторые программы Windows работают безукоризненно, другие – с небольшими проблемами, а третьи – не запускаются вовсе. Казалось бы, после десяти лет разработки любая программа должна работать идеально, однако в случае с *Wine* есть два всегда актуальных препятствия:

1 **Недокументированные возможности** Windows *Wine* воссоздает в Linux программный интерфейс приложения (API) Windows, но поскольку многие системные вызовы в Windows недостаточно описаны в документации или совсем не документированы, разработчикам *Wine* приходится воспроизводить их функционал самостоятельно.

2 **Новые версии Windows** В далеком 1993 году разработчикам нужно было обеспечивать совместимость с Windows 3.1. Через несколько лет им было нужно эмулировать API Windows 95. Каждый раз, когда полная совместимость с текущей версией Windows почти достигнута, выходит новая версия, и все начинается сначала...

Таким образом, лучшее, что вы можете сделать – это установить *Wine* и попробовать запустить некоторые Windows-приложения в Linux с его помощью. Вы увидите, что одни программы работают прекрасно, другие – нет, но если обновлять *Wine* регулярно (новый релиз выходит чуть ли каждую неделю), то результат не заставит себя ждать. На следующих страницах мы рассмотрим инсталляцию *Wine* и попробуем запустить некоторые программы.

Установка

Как мы уже говорили, *Wine* – это программа, которая предоставляет «слой совместимости» для запуска приложений Windows в Linux. На самом деле, это программа и набор библиотек, эмулирующих библиотеки DLL в Windows. К счастью, ее установка очень проста. Правильнее всего использовать менеджер пакетов вашего дистрибутива; установите все пакеты, найденные по запросу '*Wine*' (обычно это **wine**, **winelib** и/или **winetools**). Если вы не знаете что делать или версия *Wine* ниже 0.9.1, возьмите последнюю версию программы из раздела **Система** на второй стороне нашего DVD.

Например, если вы используете Ubuntu, скопируйте файлы **.deb** с нашего DVD в свой домашний каталог и выполните команду

```
sudo dpkg -i *.deb
```

Система запросит пароль пользователя, затем *Wine* будет установлен. Если ваш дистрибутив использует RPM, скопируйте файлы .rpm в свой домашний каталог и выполните команды

```
su
rpm -ivh *.rpm
```

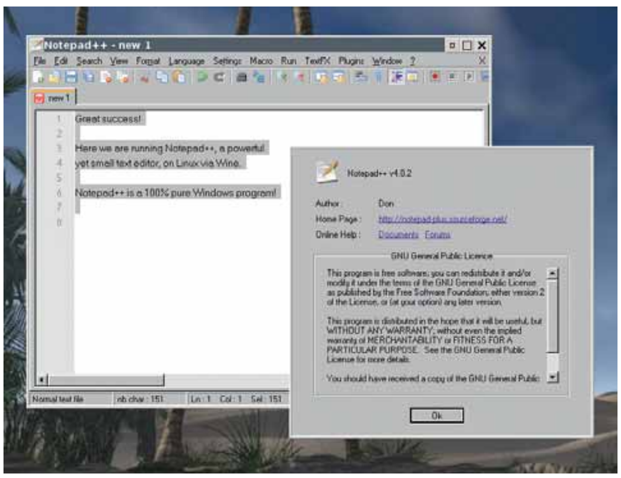
По запросу системы введите пароль пользователя root, и программа будет установлена. Еще одним вариантом является сборка *Wine* из исходных текстов. Предварительно убедитесь в том, что в системе установлены компилятор GCC и пакеты разработки для X-сервера. Скопируйте файл **wine-0.9.39.tar.bz2** из раздела **Система/Wine** нашего DVD в свой домашний каталог и выполните команды

```
tar xfvj wine-0.9.40.tar.bz2
cd wine-0.9.40/
./configure
make depend && make
```

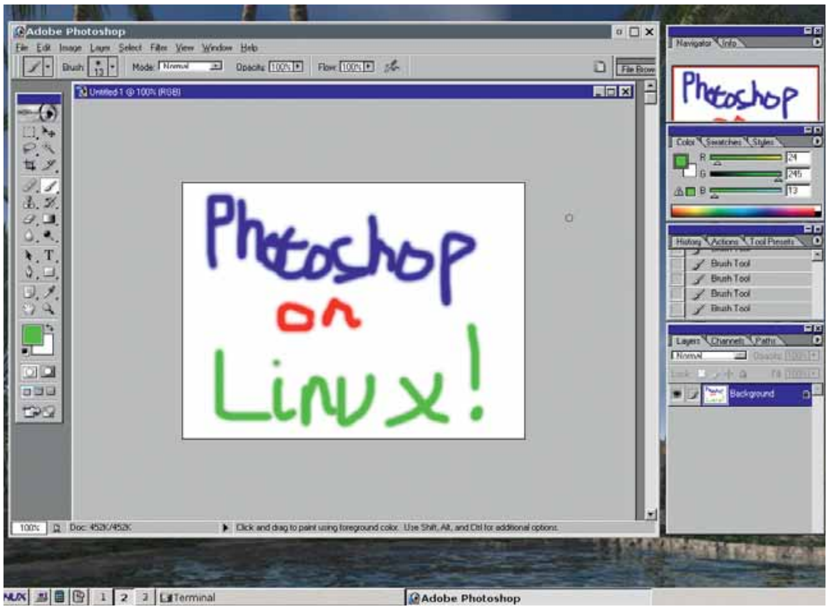
Если во время выполнения команды **./configure** появились сообщения о неразрешенных зависимостях, установите недостающие инструменты разработчика с помощью менеджера пакетов. Например, если появилось сообщение об отсутствии заголовочных файлов X-сервера (Can't find the X development headers), поищите пакеты вроде 'xorg' или 'x11' и установите те из них, названия которых заканчиваются на '-dev' или '-devel'. После этого снова выполните команду **./configure**.

Используйте команду **sudo bash** (в Ubuntu) или **su** (в других дистрибутивах) для начала сеанса работы под пользователем root, затем выполните команду

```
make install
```



› **Проверим *Wine*: *Notepad++* запускается, значит, Wine успешно установлен, и можно пробовать другие приложения..**



› **Когда-то для этого требовались часы упорной работы, а сейчас мы можем запускать известные программы с помощью нескольких команд.**

Программа *Wine* и ее библиотеки будут установлены в каталог **/usr/local**. Поздравляю – все готово! Если вы добавили *Wine* с помощью менеджера пакетов, он может оказаться в каталоге **/usr**, но вам не следует об этом беспокоиться.

Как это работает

Теперь давайте протестируем свежую версию *Wine*, запустив какое-нибудь простое приложение. На второй стороне нашего диска в разделе **Система/Wine** вы найдете файл **notepad-install.exe**. Скопируйте его в свой домашний каталог и введите в окне терминала следующую команду

```
wine notepad-install.exe
```

На экране промелькнут несколько строк отладочного текста, затем появится окно установщика *Notepad++*. Это текстовый редактор для Windows, возможности которого не идут ни в какое сравнение с *Notepad* от Microsoft. Кроме того, его маленький размер в данном случае имеет значение. Итак, программа для Windows идеально работает »

Что можно запускать?

Вообще говоря, лучше использовать более старые версии программ, потому что свежие релизы могут работать с кучей ошибок или не запуститься совсем. Поддержка игр, использующих DirectX, постоянно улучшается, поэтому если программе хотя бы года три, никаких проблем, скорее всего не будет. Вот список наиболее популярных приложений, совместимых с *Wine*:

» **Microsoft Office.** *Microsoft Office 2000* работает отлично, с *Office XP* есть некоторые проблемы, а о поддержке *Office 2007* сейчас даже не думают. *Word XP* работает отлично, хотя и медленно.

» **Internet Explorer.** Даже не думайте об использовании седьмой версии браузера, если не знаете, как с помощью *Wine* убедить установщик в том, что он запущен на подлинной системе Windows. Версию 6.0 можно использовать для тестирования создаваемого вами web-сайта.

» **Adobe Dreamweaver.** Восьмая версия, один из последних релизов *Dreamweaver*, работает вполне прилично, хотя операции копирования и вставки могут вызвать ошибки. Чтобы установить *Dreamweaver*, может потребоваться несколько попыток.

» **Adobe Photoshop.** При работе с версиями 7 и CS (8) проблем почти нет. В версии CS2 иногда возникают ошибки с расширениями и клавишами быстрого доступа.

» **Quicken.** Хотя в версии 2002 года не очень хороши шрифты и есть проблемы с печатью, в целом пакет работает стабильно по сравнению с другими релизами.

» **Apple iTunes.** Не стоит пробовать версию 7.0. Версии 4.9 и 6.x вполне работоспособны, хотя есть проблемы с передачей файлов и воспроизведением треков. Сейчас поддержка iTunes оставляет желать лучшего.

» **Half-Life 2.** Работает очень хорошо, хотя число кадров в секунду ниже, чем в Windows, и есть проблемы с графикой.

» в Linux! Как вы уже поняли, для запуска программы Windows в Linux используется команда **wine <program.exe>**.

После завершения установщика запустится редактор *Notepad++*. С его помощью мы постараемся понять, как работает *Wine*. Начнем с открытия файла (пункт меню File > Open). Нетрудно заметить, что *Wine* отображает папку **My Documents** на домашний каталог пользователя, а диск **Z**: – на корень файловой системы (**/**). Но куда указывает диск **C**:?

При первом запуске *Wine* создает в вашем домашнем каталоге папку **.wine**. В ней располагаются файлы реестра, в которых хранятся настройки программ, и подкаталог **drive_c**. Он и выступает в роли диска **C**: для приложений Windows, запускаемых с помощью *Wine*. Здесь же располагаются файлы установленных нами программ. Так, в **drive_c** можно обнаружить каталог **Program Files**, а в нем – папку **Notepad++**, в которой находятся файлы только что установленного текстового редактора.

Итак, диск **Z**: указывает на дерево файловой системы Linux, и мы можем обращаться к файлам ОС Linux из программ Windows, а диск **C**: – на каталог, который содержит файлы установленных программ (это подкаталог **drive_c** каталога **.wine** в домашнем каталоге пользователя). Если вы захотите запустить *Notepad++* снова, используйте команду **wine <program.exe>** следующим образом:

```
wine .wine/drive_c/Program\ Files/Notepad++/notepad++.exe
```

То же самое относится и к другим программам. После установки приложения найдите его исполняемый файл в папке **.wine/drive_c**. Затем вы сможете запустить программу. Для этих целей можно также приспособить ярлык на рабочем столе. Если на компьютере установлена Windows, можно зайти в ее каталог (он обычно находится в директории **/media** или **/mnt**) и попробовать «на зуб» стандартные программы. Например, для запуска классической игры *Солитер* используется команда [**/media/hda2** надо заменить на точку монтирования в вашей системе, – прим. ред.]

```
wine /media/hda2/windows/system32/sol.exe
```



Пример: Photoshop

Вам уже хочется запустить свои любимые приложения? Но, как мы уже говорили, некоторые программы работают отлично, а другие – с большим трудом. Поэтому остается только пробовать, скрестив пальцы. База данных приложений *Wine* (**http://appdb.winehq.org**) содержит список из сотен поддерживаемых программ, для каждой из которых указана «совместимость». Там вы сможете найти необходимую программу. Самые популярные программы описаны во врезке «*Что можно запускать*?» на стр. 41.

Сейчас мы установим *Adobe Photoshop 7*. Конечно, это не самая последняя версия известного графического редактора, но она очень хорошо работает в *Wine*. Скопируем установочные файлы с дистрибутивного диска в папку **Photoshop** в домашнем каталоге. Запустим терминал и выполним следующие команды

```
cd Photoshop
wine Setup.exe
```

Появится окно инсталлятора, который проведет нас через все этапы установки. Он скопирует файлы *Photoshop* в виртуальную папку Windows **.wine/drive_c** и добавит информацию о нем в файлы реестра (они также расположены в папке **.wine**). Как только инсталлятор завершит работу, мы сможем запустить *Photoshop* командой

```
wine .wine/drive_c/Program\ Files/Adobe/Photoshop\ 7.0/photoshop.exe
```

Вот и все! *Photoshop* работает в Linux. И очень неплохо: некоторые диалоги долго открываются, и окно с инструментами появляется сразу на всех виртуальных рабочих столах, но в остальном – очень удобно. Однако мы можем сделать еще кое-что...

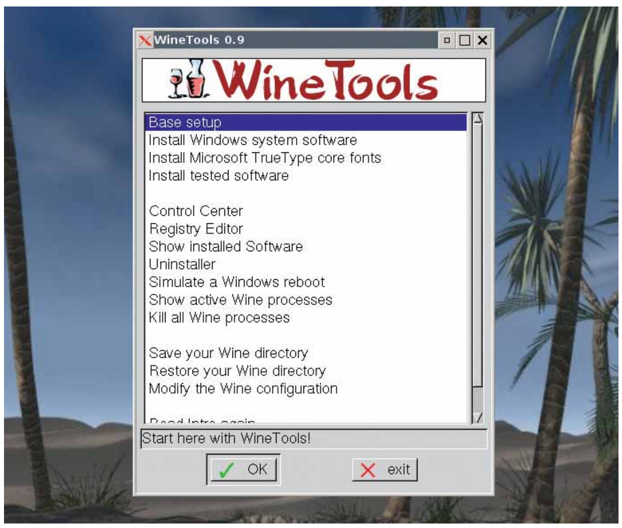
Советы по настройке

Закройте все запущенные в *Wine* программы и введите **winecfg** в окне терминала. Появится окно Windows-программы, позволяющей задавать различные настройки *Wine*, повышающие совместимость и производительность. Мы изучили советы других пользователей по оптимизации запуска *Photoshop* 7 (их можно найти на странице **http://appdb.winehq.org**) и хотим поместить все панели инструментов *Photoshop* в одно рабочее пространство. Для этого выберем вкладку Графика (Graphics), затем Эмулировать виртуальный рабочий стол (Emulate A Virtual Desktop) и установим его разрешение в 1024 x 768.

Запустим *Photoshop* снова, и voilá – он в отдельном окне! Мы больше не увидим следов его панелей инструментов после переключения рабочего стола. Попробуйте запустить другие программы, и вы оцените все удобство *Winecfg* – раньше на правку конфигурационных файлов приходилось тратить кучу времени, а сейчас практически все настройки можно изменять с помощью графического интерфейса.

На вкладке Приложения (Applications) можно даже изменить версию Windows, которую эмулирует *Wine* (например, установить Windows 98 вместо 2000). Эта возможность особенно полезна при запуске старых программ.

Вот как выглядит оптимальный способ запуска приложений Windows с помощью *Wine*: найти рекомендации по запуску этого приложения в базе данных на сайте, посмотреть на результаты тестов для этого приложения и изменить необходимые настройки в *Winecfg* до установки



› Хотя при работе с WineTools иногда возникают проблемы, с их помощью установить и запустить известные программы очень просто.

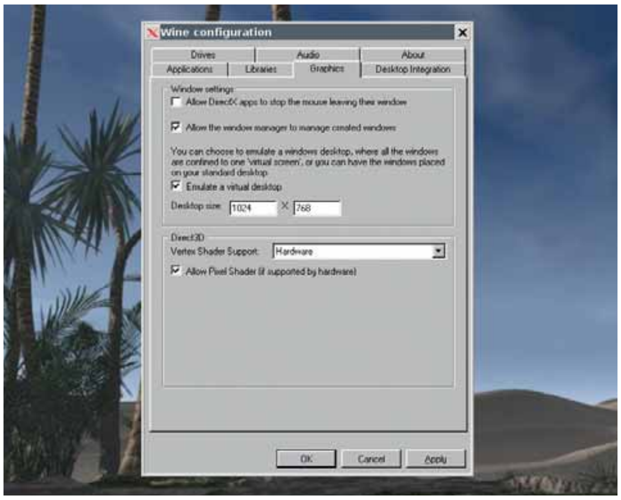
CrossOver и Cedega

Существуют также коммерческие версии *Wine*, которые используют ту же кодовую базу, но добавляют удобства – в них вам не придется открывать командную строку, чтобы запустить Windows-программу. В отличие от свободного *Wine*, эти решения фокусируются на поддержке вполне определенного набора приложений, зато достигают лучшей совместимости с ними.

CrossOver Linux от CodeWeavers (**www.codeweavers.com**) – это модификация *Wine*, оптимизированная для запуска популярных программ, среди которых *Office 2003/XP*, *iTunes* и *Lotus Notes*. Он также поддерживает расширения браузера, в том числе проигрыватели форматов QuickTime и Windows Media. Разработчики *CrossOver* концентрируют свои усилия на улучшенной (по сравнению с *Wine*) поддержке этих программ. Если вы пользуетесь только ими, то *CrossOver* Linux за $40 – прекрасная альтернатива полноценной копии Windows.

Хотя *CrossOver* поддерживает и некоторые игры, «хардкорным геймерам» гораздо лучше подойдет *Cedega* (**www.transgaming.com**). Эта программа, ранее известная как *WineX*, предназначена для запуска игр, в том числе *World of Warcraft*, *Oblivion*, *Civ IV* и многих других (полный список можно посмотреть на сайте программы).

Мы уже рассказывали о *Cedega* в этом месяце и в предыдущих обзорах, и ее производительность произвела на нас впечатление – часто невозможно понять, запущена игра в эмуляторе или в настоящей Windows. На момент написания статьи годовая подписка на *Cedega* стоила $60, и с учетом ее высоких темпов развития, это почти даром.



› *Winecfg* позволяет настроить Wine без утомительного ручного редактирования конфигурационных файлов.

или запуска программы. Если установить программу не получается и на компьютере установлены две операционные системы, можно попробовать запустить ее прямо с раздела, на котором установлена Windows (мы проделали это с *Солитером*). Раздел Windows обычно смонтирован в папке **/mnt** или **/media**.

Итак, давайте вспомним основные моменты: для запуска программы используется команда **wine <progname.exe>**, файлы виртуальной Windows хранятся в папке **.wine/drive_c** в домашнем каталоге пользователя, а для настройки параметров запуска программ в соответствии с описанием программы в базе данных *Wine* нужно ввести команду **winecfg**. Для запуска установленной программы можно не пользоваться командной строкой. Достаточно создать ярлык для исполняемого файла программы, который находится в папке **.wine/drive_c**, и прописать в нем команду для запуска программы в *Wine*, например:

```
wine /home/username/.wine/drive_c/myprog.exe
```

Небольшой совет: если после установки новой программы *Wine* перестает нормально работать, удалите папку **.wine** в своем домашнем каталоге. Она будет создан снова при следующем запуске *Wine*. Конечно, все остальные программы придется переустанавливать, но это полезно знать на случай, если ваш *Wine* «прикажет долго жить».

Установка с WineTools

Хотя установка программы из окна терминала – дело нехитрое, *WineTools* (**www.von-thadden.de/Joachim/WineTools**) использует для этого графический интерфейс, управляемый с помощью меню. Скопируйте архив **winetools-0.9jo-III.tar.gz** из раздела **Система/WineTools** на второй стороне нашего DVD на жесткий диск и введите в окне терминала следующие команды:

```
tar xfvz winetools-0.9jo-III.tar.gz
cd winetools-0.9jo-III
```

Начните сеанс работы с правами пользователя root с помощью команды **sudo bash** (Ubuntu) или **su** (в других дистрибутивах), затем введите **./install**. После этого введите **exit** для окончания сеанса root. Введите **wt**, и на экране появится первое диалоговое окно установщика. Выберите базовую установку (Base Setup), затем обновите каталог **.wine** (Update An Old Fake Windows Drive). После этого на экране появится окно базовой установки (Base Setup). В этом окне нужно щелкнуть главное меню (Main Menu).

В режиме установки протестированного ПО (Install Tested Software) можно добавлять различные приложения, которые будут загружены из Интернета или взяты с CD/DVD-диска. Например, если выбрать Acrobat Reader в меню Player, *WineTools* загрузит инсталлятор этой программы, покажет все зависимости и запустит установщик.

Обратите внимание, что в некоторых случаях *WineTools* потребуется установка *DCOM98* и *Internet Explorer* необходимой версии, это можно сделать в меню базовой установки (Base Setup). И это тот случай, когда совместное использование *Wine* и *WineTools* может вызвать проблемы – так наш компьютер завис при установке *DCOM98*, но будем надеяться, что в будущих версиях *Wine* и/или *WineTools* эта проблема будет решена. Несмотря на некоторые шероховатости, многие программы отлично работают, кроме того, *WineTools* избавляет нас от необходимости поиска нужного **.exe**-файла при загрузке программы из Интернета.

Будущее Wine

А что же Microsoft? Компания пока не сделала официальное заявление о проекте *Wine*, но службы Windows Update и Windows Genuine Advantage уже проверяют системный реестр на наличие ключей *Wine* и блокируют все обновления системы в случае их обнаружения. Для большинства из нас это не проблема – отдельные программы Windows гораздо важнее последней версии дефрагментатора диска. Тем не менее, это говорит о том, что Microsoft очень внимательно следит за развитием *Wine*.

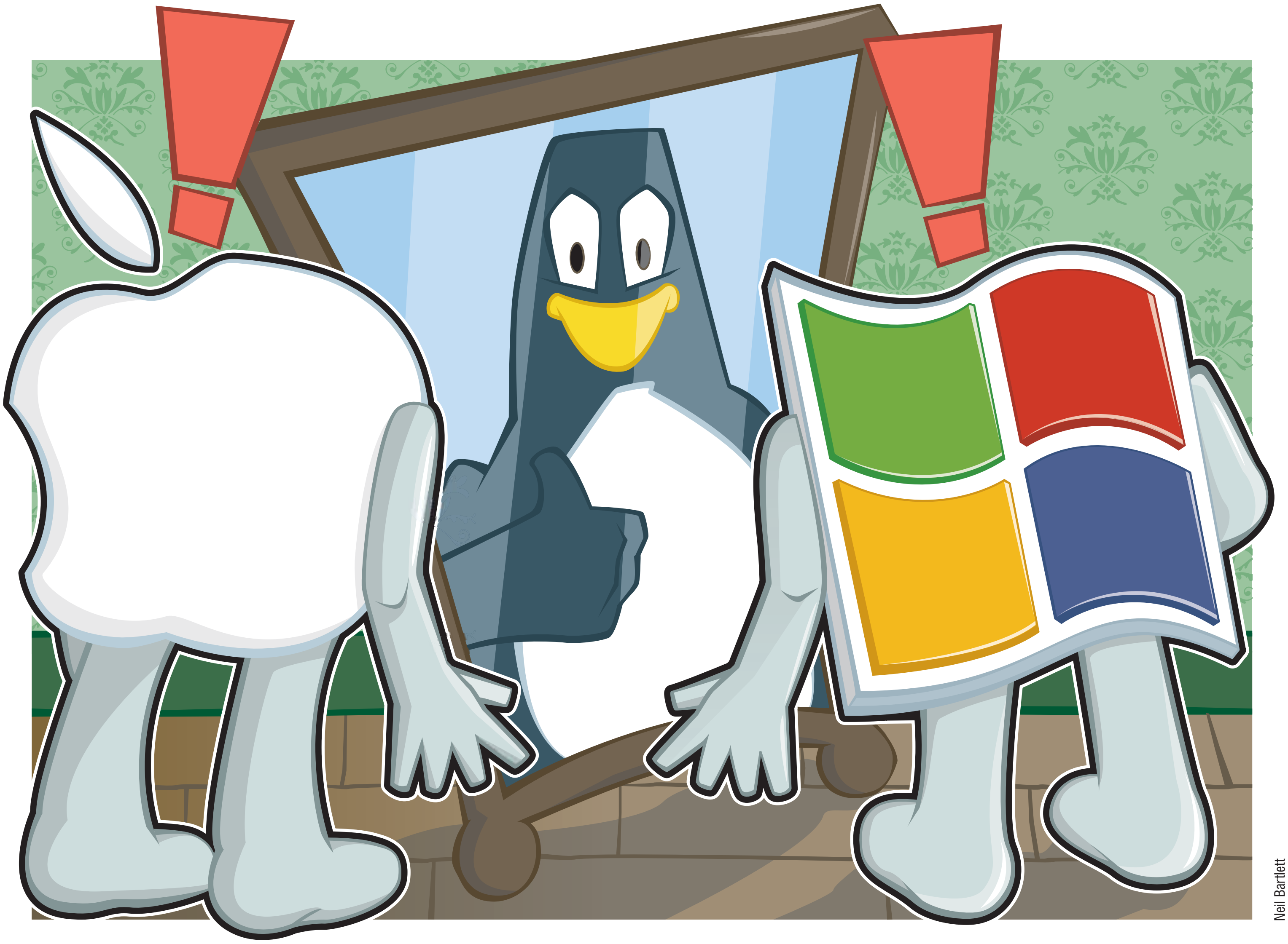


Многообещающим проектом из мира *Wine* является *Wine-Doors* (**www.wine-doors.org**), программа для Gnome, которая расширяет функционал *WineTools* и еще больше упрощает установку программ Windows в *Wine*. На момент написания статьи не было даже версии 0.1, и исходный код был доступен только через SVN, но первый релиз может появиться очень скоро. Следите за информацией на сайте проекта.

Между тем, разработчики *Wine* прокладывают дорогу к версии 1.0. Мы ждали ее многие годы, но с выпуском Windows Vista момент выпуска снова может быть отодвинут. Впрочем, в сохранении номера 1.0 для полноценной работающей версии нет ничего зазорного. Расскажите нам о том, как вы установили *Wine* – наши форумы **www.linuxforum.ru** открыты для обмена положительными эмоциями, помощи, если что-то не получается, и других вопросов по запуску разных программ Windows в *Wine*. Удачи! LXF



› С помощью *Wine* можно запускать как серьезные программы, так и игры, например, *Half-Life 2*.



Neil Bartlett

FOSS за пределами Linux

Игнорировать существование альтернативных операционных систем все еще трудновато. Но даже в бесплодной пустыне закрытой ОС встречаются оазисы, где цветет свободное ПО. За ними наблюдает **Грэм Моррисон**.

LINUX FORMAT на DVD

Для всех программ, кроме *Ardour* и *Seashore*, существуют версии под Windows и Mac.

- Ardour 0.99.3
- Audacity 1.2.6
- Inkscape 0.45.1
- Blender 2.43
- Gimp 2.2.x
- Scribus 1.3.3.x
- Seashore 0.1.9

Между двумя группами сторонников открытого программного обеспечения уже давно ведется дискуссия, в основе которой один вопрос: доступность свободного ПО на закрытых платформах способствует развитию Open Source или сдерживает его?

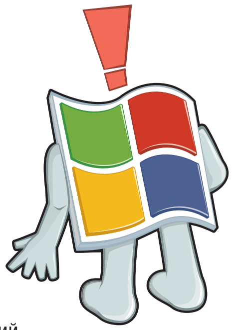
Некоторые, в их числе Ричард Столлмен, считают, что возможность запуска свободного и открытого ПО (Free and Open Source Software, FOSS) в таких операционных системах, как Microsoft Windows и Apple OS X, уменьшает число желающих попробовать Linux. Другие полагают, что доступность свободного ПО для закрытых систем дает шанс познакомиться с ним и тем, у кого иначе не было бы такой возможности. Это положительно сказывается на качестве проектов с открытым исходным кодом и их разнообразии. Хороший пример здесь – *Inkscape*. Для этой программы доступны двоичные сборки, скомпилированные для Windows, OS X и Linux, а девиз «Рисуй свободно» (Draw Freely) на сайте программы и в окне «О программе» говорит сам за себя.

К счастью, свобода заразительна. За последние годы некоторые свободные и бесплатные приложения для Windows и OS X стали незаменимыми для многих. Хотя люди и не вникают в идеологию, стоящую за бесплатными офисными пакетами, первоклассными web-браузерами или сетевыми стеками *Samba*, большее число пользователей означает лучшее тестирование, бóльшую известность и заставляет сторонних производителей поддерживать программы, которые мы используем в Linux. Если бы браузер *Mozilla Firefox* не снискал популярности, у многих сайтов до сих пор были бы проблемы с совместимостью, а формат OpenDocument не был бы одобрен как стандарт ISO. Ну, а пользователи старинной OS/2 от IBM не смогли бы запускать передовые приложения из мира открытого ПО, например, *Scribus*.

Разрабатываем свободно

Самое большое преимущество открытого ПО для закрытых платформ в том, что даже если вы не выбираете, в какой операционной системе работать (например, в офисе или в кафе), вы можете воспользоваться теми же программами, что и дома. Для многих проектов доступны версии, скомпилированные под другие ОС, и, что удивительно, некоторые программы работают в Windows или OS X гораздо лучше, чем в Linux. Многие приложения проще установить и выполнить их обновление. Однако именно Linux предоставляет разработчикам плодородную почву, на которой всходят ростки многих открытых проектов. Это сочетание средств разработки, сообщества и желания предоставить свободную альтернативу другой ОС движет вперед весь процесс развития открытого ПО. Оно также вынуждает крупные компании, такие как Apple и Microsoft, пересмотреть свое отношение к Open Source, что не может не радовать.

FOSS в Windows



От Firefox и OpenOffice.org до X11 и GNU Make.

После установки Windows многие из нас первым делом загружают свежую версию *Firefox*. Windows-редакция браузера функционально аналогична Linux-версии, и после ее загрузки и установки вы можете пользоваться *Firefox* точно так же, как делали это в Linux. По соображениям безопасности *Firefox* более предпочтителен, чем браузер, поставляемый с Windows по умолчанию. Кроме того, он содержит средства блокировки всплывающих окон и нежелательных апплетов.

Одним из наиболее важных расширений *Firefox*, которое позволяет синхронизировать закладки, историю и даже cookies между несколькими установленными копиями браузера, является Google Browser Sync. Доступны версии для Windows, OS X и Linux. Все данные пользователя связываются с его учетной записью в Google и хранятся на web-сервере. Мы бы не стали доверять генератору прибыли Google историю всей своей жизни в сети – но ограничившись только закладками, вы получите весьма полезный инструмент, особенно если на подбор информации были потрачены годы.

Вторая программа, которую необходимо установить в Windows – это *OpenOffice.org*. Подобно *Firefox*, этот набор офисных приложений вполне привычен и работает в Windows точно так же, как и в Linux. Это прекрасная альтернатива другим офисным пакетам, которые стоят больших денег.

Поскольку *Firefox* и *OpenOffice.org* основаны на кроссплатформенных библиотеках, их проще всего запустить в Windows. С другими приложениями, основанными на различных технологиях Linux, начинаются сложности. Одним из наиболее известных примеров является *GIMP*, прекрасный графический редактор.

GIMP в Windows

GIMP использует библиотеку *GTK 2*, которая настолько глубоко интегрировалась в рабочий стол Linux, что сама стала им. Многие функциональные возможности GNOME реализуются именно *GTK*. К счастью, библиотека портирована как под Microsoft Windows, так и под Apple OS X. Она необходима для работы *GIMP* на этих системах.

В разных операционных системах библиотека *GTK* используется по-разному. В Windows *GTK* «встроена» в приложение и обычно устанавливается в виде библиотеки DLL в каталог программы. Следовательно, к библиотеке невозможен общий доступ, но нет и конфликта версий, когда различные приложения требуют *GTK* разных версий. Установщик *GIMP* от Джерни Симончича [Jernej Simoncic] содержит библиотеку *GTK 2* для Windows (5,5 МБ) и исполняемые файлы *GIMP* (2 МБ). Установка *GTK* и *GIMP* производится из единого **exe**-файла, загрузить который можно с сайта SourceForge.

Иногда может потребоваться установка одного или нескольких дополнительных пакетов. Например, вместе со *Scribus*, лучшей настольной издательской системы с открытым исходным кодом, потребуется установить *Ghostscript* для импорта и печати в форматы PostScript и EPS. Инсталлятор автоматически загрузит необходимый файл и распакует его в каталог программы. После этого *Scribus* будет работать точно так же, как и в Linux.

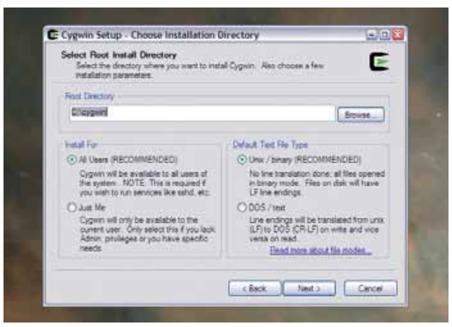
Для работы Windows-версии *Pidgin*, программы, ранее известной как *Gaim*, тоже необходим *GTK*. Кстати, библиотека *Pidgin* используется и другой известной [свободной, – прим. ред.] программой обмена сообщений Windows – *Miranda*. Еще два полезных приложения – аудиоредактор *Audacity* и векторный графический редактор *Inkscape* – также очень популярны и имеют мало конкурентов.

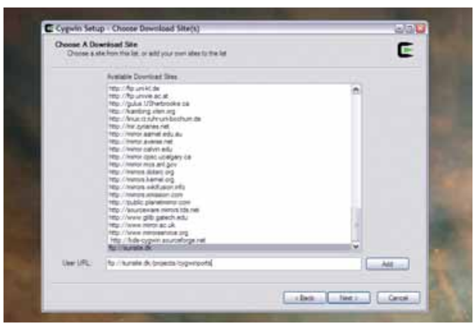
Дополнительно можно загрузить *VST Enabler* – компонент *Audacity*, который позволяет использовать в программе любой из сотен свободно распространяемых звуковых эффектов VST. Это большое преимущество по сравнению с версией для Linux. Между тем, разработчики *Inkscape* заявили, что они хотят сделать *Inkscape* доступным всем пользователям – независимо от платформы – и вкладывают большие усилия в унификацию пользовательского интерфейса. »

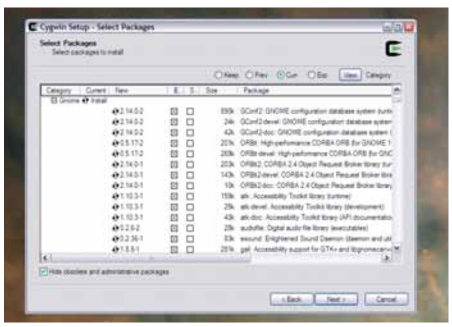
› Многие открытые приложения устанавливаются в Windows одним щелчком мыши.



Шаг за шагом: Запускаем GNOME в Windows







1 Устанавливаем Cygwin ›

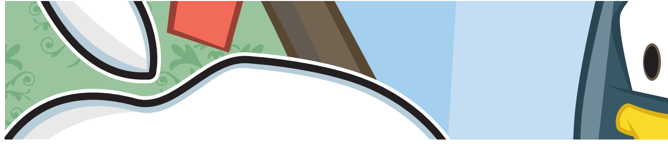
Cygwin транслирует системные вызовы POSIX в Windows и, таким образом, создает окружение для работы Linux-программ. Начнем с загрузки и запуска исполняемого файла *Cygwin*.

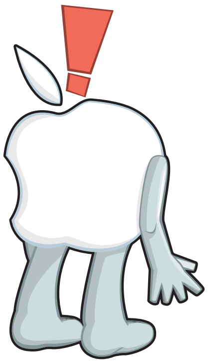
2 Определяем список пакетов ›

Выберите ближайший репозиторий с пакетами *Cygwin*. Дополнительные репозитории для сборок GNOME и KDE можно указать вручную. После этого программа установки загрузит список пакетов.

3 Загружаем GNOME

Выберите из списка необходимые пакеты и нажмите кнопку Далее (Next). *Cygwin* загрузит около 1 ГБ файлов, затем на рабочем столе будет создан ярлык для запуска GNOME.





FOSS в OS X

От *Celestia* и *Stellarium* до *Fink* и Macports.

Работать с открытым ПО в Microsoft Windows более или менее просто. Свободные приложения загружаются и устанавливаются точно так же, как и обычные программы, и пользователь привыкает к ним, просто освоив новый интерфейс. В Apple OS X все обстоит несколько по-другому.

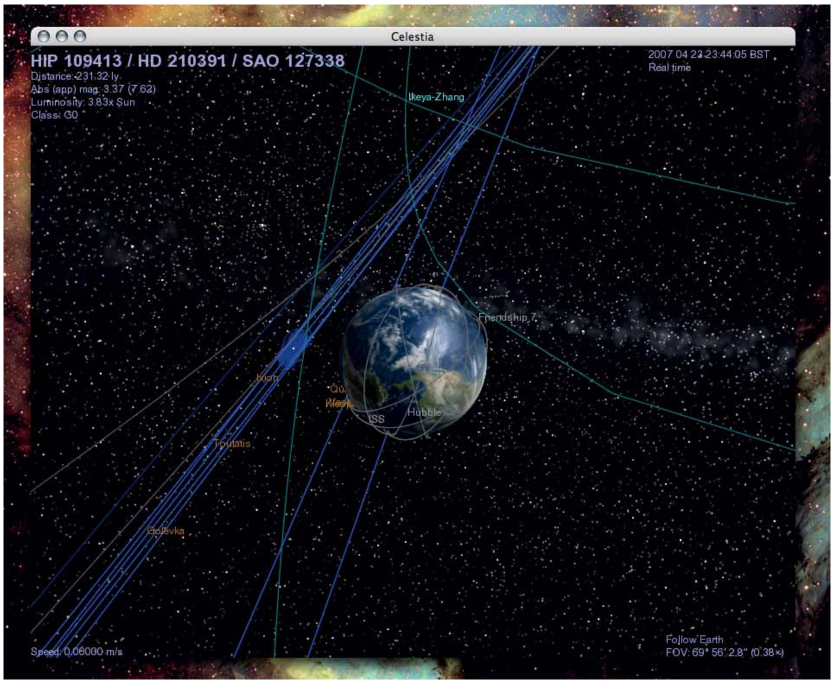
OS X – изящная операционная система, которая использует очень строгий и тщательно проработанный подход к интерфейсу пользователя. Хотя GNOME и стремится копировать его, современным открытым программам очень не хватает изящества и продуманности дизайна приложений OS X. Но забудем на мгновение об эстетической стороне вопроса, ведь OS X – это максимально близкий аналог Linux, который можно купить за деньги.

Больше X, чем X11

Поскольку Apple OS X является Unix-подобной операционной системой, эта платформа отлично подходит для запуска открытого ПО. OS X включает бесплатный инструментарий разработчика, и Apple даже использует свободные компоненты в ключевых областях своей системы. Мы говорим о сетевом стеке *Samba*, web-сервере *Apache* и KHTML.

Однако из-за различий между OS X и Linux ситуация на Mac'ах несколько усложняется. Многие открытые приложения, включая те, что используют *GTK*, требуют X11-совместимой графической подсистемы. В Linux этот слой реализуется *X-сервером*, и мы рассматриваем *X Window System* как данность. А в OS X используется собственный движок для отрисовки графики. Тем не менее, вы можете установить *X-сервер* и тут. Необходимые архивы можно загрузить с web-сайта Apple или найти на дисках, которые поставлялись вместе с вашим Intel Mac.

Процесс займет продолжительное время, поскольку OS X должна пройти все этапы процедуры установки. Если вам повезло инсталлировать OS X «с нуля», сразу же выберите *X-сервер* наряду с другими пакетами. Если же вы тяготеете к продуктам сообщества, попробуйте *XDarwin*, базирующийся на исходном коде *XFree86*. Совместимость с X11 высоко ценится разработчиками – изменения в исходном коде программ, необходимые для их компиляции под *X-сервер* в Apple OS X, незначительны, а функциональные возможности приложений точно такие же, как и в Linux. Наберите **man startx** в окне терминала, и вы поймете, что мы имеем в виду.



› Если бы свободные астрономические программы, такие как *Celestia*, были платными, их вряд ли ожидал бы большой успех.

Кроссплатформенный

Запишите портированные версии этих открытых программ на USB-брелок – и будете чувствовать себя в Windows и OS X, как дома:

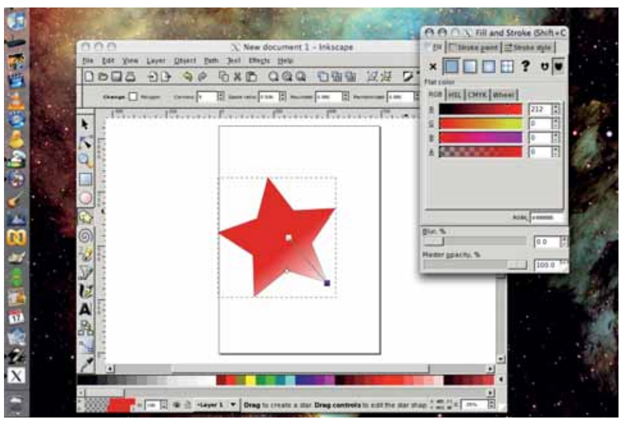
- » *Gimp* http://gimp-win.sourceforge.net
- » *Blender* www.blender.org
- » *Inkscape* www.inkscape.org
- » *Scribus* www.inkscape.org
- » *VLC* www.videolan.org
- » *Audacity* http://audacity.sourceforge.net
- » *Firefox* и *Thunderbird* www.mozilla.com
- » *OpenOffice.org* www.openoffice.org

Inkscape и Scribus

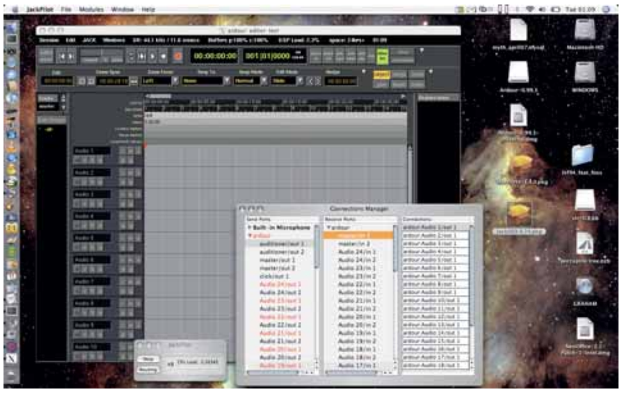
Благодаря тому, что приложения портируются достаточно просто, в OS X вы сможете пользоваться бо́льшим количеством открытых программ, чем в Windows. *GIMP* и *Inkscape* являются достаточно крупными проектами, и для каждого из них есть сборки и установочные пакеты для OS X. С другой стороны, *Scribus* предъявляет особые требования – ему нужен процессор PowerPC, а производительность на компьютерах Intel будет гораздо ниже, потому что на них будет использоваться встроенный в OS X эмулятор PowerPC Rosetta.

Несмотря на это, портированные в OS X приложения, которые используют *X-сервер*, ведут себя «как положено». Они даже выглядят как приложения OS X, за исключением одной детали: для работы с буфером обмена, копирования, вставки и других операций используются другие сочетания клавиш (например, Ctrl+X вместо Apple+X для команды «вырезать»). Кроме того, теряется интеграция приложений с рабочим столом, поскольку сами приложения не знают, в какой операционной системе они выполняются. Например, вы не сможете перетащить файлы на палитру инструментов *GIMP*, как в Linux или Windows – вам придется воспользоваться маленьким значком в док-панели.

Многие приложения используют достаточно простой графический интерфейс, и для них не нужен *X-сервер*. Эти программы «просто будут работать». Среди них *VLC* – пожалуй, самый универсальный проигрыватель мультимедиа, и два лучших виртуальных планетария: *Stellarium*



› Версии *Inkscape* для OS X необходим *X-сервер*.



› **Звуковой редактор *Ardour* использует версию Jack из OS X.**

и *Celestia*. Эти приложения безупречно работают в OS X и Windows и являются хорошей рекламой для Open Source. Благодаря тому, что Trolltech выпустила версию библиотеки *Qt* под лицензией GPL, в OS X доступны все возможности *MythTV* – программы, разработка которой под Windows потребовала бы нечеловеческих усилий из-за строгих правил использования библиотек, конфигурационных файлов и каталогов, применяемых в Windows. Trolltech смогла в максимальной степени использовать схожесть Linux и OS X, чтобы упростить процесс сборки приложения из оригинального исходного кода.

X-терапия

Для некоторых разработчиков дополнительный слой, который появляется в OS X с установкой *X-сервера*, стал дополнительной проблемой. В результате появились проекты, реализующие версии открытых приложений, «родные» для OS X. Самый известный из них, *NeoOffice*, представляет собой OS X-редакцию *OpenOffice.org*. Сразу после выхода очередного релиза *OpenOffice* все «внутренности» программы «переносятся» в приложение OS X. Так, *NeoOffice* располагает панель меню вверху экрана, а также использует шрифты и драйвера принтера из OS X. Для запуска *NeoOffice* не нужен *X-сервер*. Однако версия *NeoOffice* всегда отстает на одну или две от текущей версии *OpenOffice.org*, например, актуальный релиз *NeoOffice* – 2.1 beta.

Разработчики *Seashore* пытаются снять зависимость от *X-сервера* у другого популярного открытого приложения. За этим непривлекательным названием скрывается проект, реализующий «родную» для OS X версию *GIMP*. В отличие от *NeoOffice*, в *Seashore* не стараются перенести в проект каждую новую возможность своего открытого собрата. Например, в нем нельзя управлять рендерингом шрифтов. Но наиболее важные функции (палитры слоев и кистей, а также фильтры, которые нам так нравятся в *GIMP*) сохранены.

Некоторые приложения работают в OS X значительно лучше, чем в Linux. Среди них *Ardour*, уникальный многодорожечный звуковой редактор. Для работы *Ardour* необходимы X11 и аудиосервер Jack. Версия Jack для OS X устанавливается одним пакетом, так что эта процедура здесь в тысячу раз проще, чем в Linux. И поскольку все Mac'и используют один и тот же Core Audio API, Jack безо всяких потерь внедряется в звуковую подсистему. Пользоваться одним из самых мощных из свободно распространяемых звуковых редакторов в OS X гораздо проще. Надеемся, что это сделает и его установку в Linux еще проще.

Для достижения лучшей совместимости и большего выбора свободных программ в OS X существуют два проекта, которые позволяют портировать приложения Linux и запускать их без особых усилий. Это Fink и Macports. Оба они организуют в OS X Linux-подобную среду разработки и используют менеджер пакетов для установки приложений и исходных текстов в Linux-подобное рабочее пространство. Fink даже устанавливает утилиту управления пакетами *apt-get*, которая работает точно так же, как и в Debian (Macports использует для тех же целей команду port). В результате вы максимально приближаетесь к работе в Linux на компьютере Mac без необходимости использовать двойную загрузку или виртуальную машину. Этот вариант вполне жизнеспособен, если по каким-то причинам вы вынуждены работать в OS X.

С правильно сконфигурированной средой разработки вы сможете загружать свежие версии приложений и компилировать их совместно с библиотеками Fink. Таким образом, вы сможете запускать приложения, которые не были портированы под OS X, например, *GnuCash*. LXF



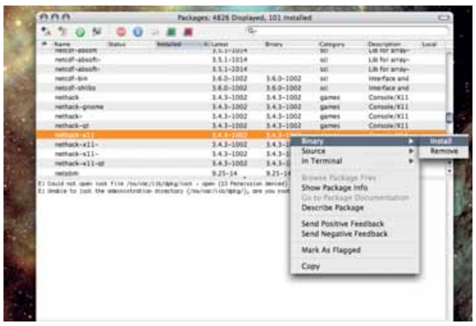
› **На дешевом Mac Mini можно пользоваться *MythTV* – идеальной программой для просмотра ТВ. Компилировать ее не придется.**

Шаг за шагом: Установка Nethack с помощью Fink



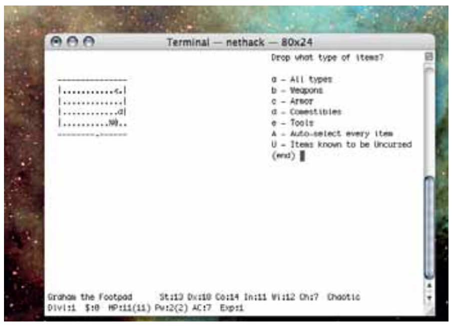
1 Находим программу

Fink включает менеджер пакетов с графическим интерфейсом, наподобие Synaptic. Он позволяет обновлять установленные пакеты и добавлять новые путем выбора их из списка или поиска файлов.



2 Выбираем метод установки

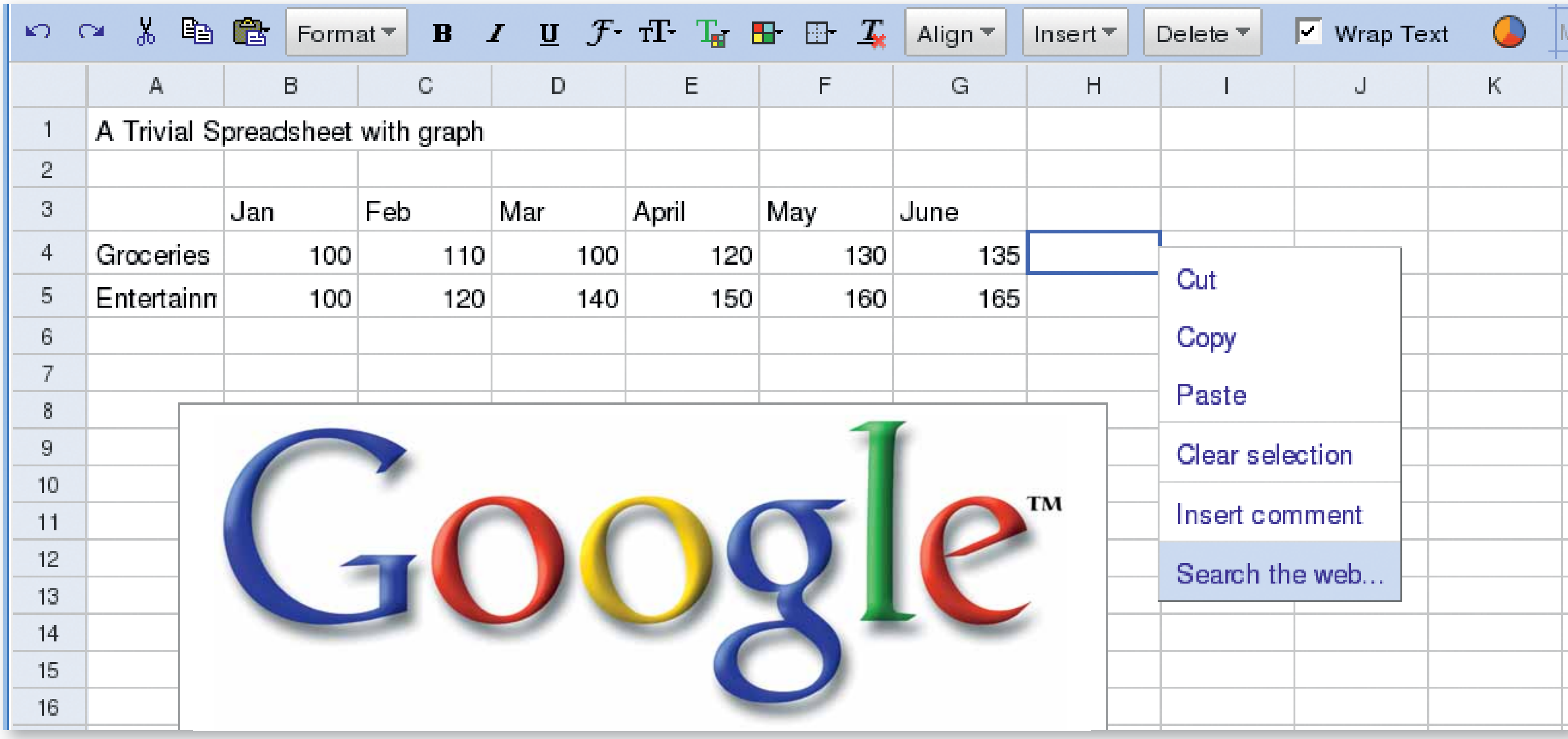
Каждый из выбранных пакетов может быть установлен либо в виде двоичного файла, либо собран из исходных текстов. Зависимые пакеты будут загружены из репозиториев в Интернете и установлены автоматически.



3 Запускаем и готово!

После установки приложение можно запустить из окна терминала. В нашем примере, для этого используется команда **nethack**.







Пусть Google поработает

Инструменты на базе web, может, и низки по своим функциям, зато высоки по минимализму; а уж по части совместной работы им просто нет равных, утверждает **Маркус Грин**.

Давным-давно, на заре развития web, Марк Андрессен [Marc Andressen] сделал громкое заявление о том, что *Netscape* превратит MS Windows в «набор слабо отлаженных драйверов устройств». Подразумевалось, что браузер освободит нас от тирании операционной системы. Что ж, возможно, все получилось не вдруг, но по крайней мере рассвет этого дня уже заметен, благодаря последним web-приложениям и самому примечательному из них – Google Docs & Spreadsheets.

Эти инструменты Google не являются свободными, и их код не открыт, но Google интенсивно использует свободное ПО и вносит большой вклад в такие проекты, как *Mozilla*, *Wine* и *OpenOffice.org*. Вдобавок мини-пакет *Docs & Spreadsheets* представляет самую крупную установку инструментов, совместимых с форматом файлов *OOo*. Тем самым база для **ODT**, формата сравнительно скромно представленного офисного пакета, расширилась за счет инструментария одного из самых влиятельных создателей программного обеспечения на планете. Если вы привязаны к *OOo*, но считаете небесполезным иметь доступ к мобильным офисным инструментам через такой сервис, как Google, узнайте на следующих страницах, какие из ваших любимых функций поддерживаются.

Docs & Spreadsheets принадлежат к относительно новой категории ПО, зависящей от Ajax, чье полное название (Asynchronous JavaScript and XML) напоминает о том, что это – хитроумное использование JavaScript. JavaScript, конечно же, встроенная в браузер технология: код загружается вместе с web-страницей и исполняется локально. Исторически, проблема JavaScript – совместимость между разными браузерами: код, работающий в *Netscape*, мог отказать в *Internet Explorer*, и наоборот. Создание кросс-браузерного JavaScript сейчас стало намного проще, но все же лучше оставайтесь с *Firefox*. Да, мы в курсе, что *Opera* и *Konqueror* почти полностью совместимы с ним, однако, к несчастью, это то же самое, что быть почти беременной. Если вы попробуете использовать инструменты Google в *Konqueror* или *Opera*, то получите предупреждение, что попробовать-то можно, но поддержка, вероятно, не сработает.

Мыслить по-новому

Интерфейс Web-Ajax имеет наследственные ограничения – и Google даже не пытался сквозь них пробиться. Но считать web-инструментарий Google соперником *Microsoft Office* – все равно, что считать телевидение соперником радио: это просто совершенно иной подход. Почти наверняка роль полновесных приложений для рабочих столов будет продолжать расти. Например, электронная почта, основанная на web, в основном используется частными лицами, поскольку корпорации неохотно отдают e-mail на откуп другим компаниям по соображениям конфиденциальности и безопасности. Но преимущества web-инструментов сделают их незаменимыми для громадного количества людей.

Одно из ключевых свойств, стимулирующих их использование – легкость и простота взаимодействия и публикации документов в Web. Для большинства людей смысл создания таблиц и документов состоит в том, чтобы показать их другим людям, ради комментариев и взаимодействия с ними. В наши дни эта задача из тяжкой поденщины превратилась в пару щелчков по клавишам.

В будущем люди будут оглядываться на цикл «Print, Post, Return» с тем же изумлением, с каким инопланетяне Cadbury Smash наблюдали за процессом чистки картошки и превращения ее в пюре. Основанная на web публикация – главная альтернатива традиционному подходу. Пока что налицо существенные ограничения на действия, которые можно производить с содержимым, но в какой-то мере эта проблема будет решена благодаря широкополосному доступу и грамотному программированию.

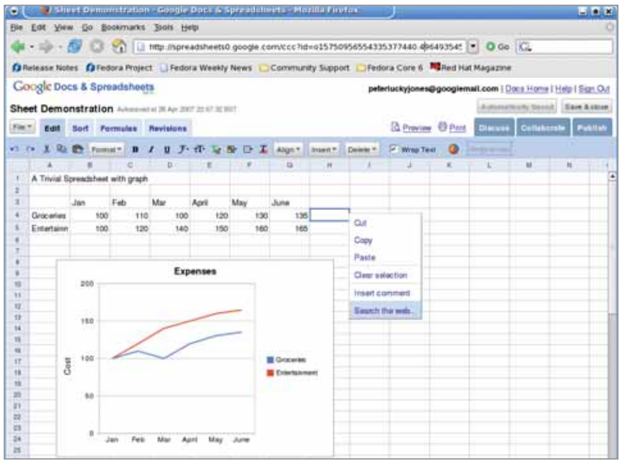
Google Spreadsheets

Сохраняются ли в этом онлайн-приложении графики и форматирование?

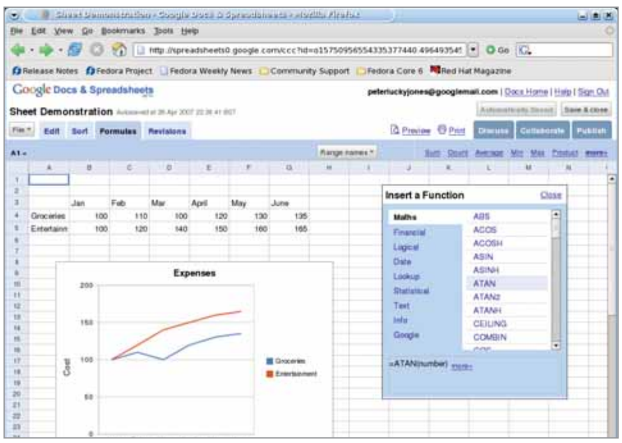
Ассортимент инструментов Google оказывается больше и больше всякий раз, когда вы обращаетесь к нему: в нем появились календарь, сетевой фотоальбом *Picasa* и *Gmail*, но сейчас нас интересуют офисные приложения, начиная с модуля таблиц. С помощью *Google Spreadsheets* вы можете загружать документы или создавать их на лету через web-интерфейс. Система распознает файлы *Calc* от *OpenOffice.org* и *MS Excel* и пытается сохранить их структуру. Мы говорим «пытается», поскольку некоторые функции теряются, а некоторые реализованы лишь частично. Сохранается самое основное из тривиальной таблицы: тип шрифта – жирный, курсив, подчеркивание; простейшие формулы, такие, как сложение и арифметические и основные функции. Однако поддержка чуть более сложных функций уже нуждается в заплатках.

Google Spreadsheets не поддерживает макросы (VBA), условное форматирование или именованные диапазоны, и обладает неполной функциональностью для графиков, позволяющей создавать таковые в модуле страницы, но, похоже, лишающей готовых графиков подгружаемые таблицы. Ограничения на графики – солидный недостаток, ибо визуальное представление данных является невероятно ценным инструментом, помогающим лучше понять их смысл.

Excel и *OpenOffice.org* поддерживают весьма полезную функцию, именуемую условным форматированием, с помощью которой, например, можно выделить все отрицательные величины красным цветом. Увы, при загрузке в модуль таблицы Google условное форматирование теряется. Возможно, неким признаком незрелости программы является отсутствие поддержки создания именованных групп ячеек. Если работаешь с чем-то большим, нежели тривиальная таблица, именованные группы делают формулы куда понятнее. А именно, вместо использования адресов ячеек можно обозначить их диапазон удобочитаемым ярлыком. Вместо формулы =SUM(C10:C12) назовите группу ячеек C10:C12 именем наподобие houserent («квартплата») и измените формулу на =SUM(houserent). В *Google Spreadsheets* именованные группы создавать нельзя, но если вы загрузите таблицу из *Excel* или *OpenOffice.org*, имена сохранятся, и использующие их формулы продолжают работать.



› Недавно добавленная функция создания диаграмм пока что довольно ограниченна.



› Вставка формулы в таблицу из списка функций.

Где мое автозаполнение?

Автозаполнение означает, что вы можете ввести величины из некой серии в две ячейки, а затем перетащить в другие ячейки, и они будут заполняться автоматически, используя последовательность. Например, в *Excel* или *OOo*, введя в одну ячейку 'Jan' (янв), а в следующую – 'Feb' (фев), вы можете перейти в соседние ячейки, и они будут заполняться так: 'Mar' (март), 'Apr' (апр), 'May' (май) и т.д. Это работает для нескольких видов последовательностей, например, дней недели или цифр. Модуль таблиц Google не поддерживает этот вид автозаполнения. Чтобы вчерне решить этот вопрос, вам, возможно, придется открыть модуль *Calc* в *OpenOffice.org*, сгенерировать автозаполнение ряда величин, а затем вставить их в таблицу Google.

Google Spreadsheets поддерживает сортировку данных в одной колонке, но не поддерживает фильтров. В стандартных электронных таблицах фильтры позволяют легко находить данные. Обычно их используют для поиска данных по одному критерию – «показать всех клиентов, в колонке адреса которых стоит Йорк» – но самые современные таблицы позволяют легко создавать достаточно сложные запросы, типа «показать всех клиентов с фамилией Смит, проживающих в Йорке или Лидсе». Это значительный пробел в наборе инструментов, который скорехонько заметят даже те, кто пользуется таблицами лишь эпизодически.

»

Зачем нужны тэги?

Оба инструмента Google обладают одной любопытной особенностью: в них нет сохранения документов в структуре каталогов – в наше время все делается через тэги. Тэги – это сейчас горячая новость, потому что они используются на сайтах социальных сетей для общения, и они замечательно удобны как способ организации документов. Это отлично: иерархическая организация документов – не священная корова; на самом деле, она создает немалые ограничения, поскольку файл может находиться только в одной директории, а ведь его, возможно, нужно отнести более чем к одной категории. Используя тэги, можно приписать файлу более одного понятия. Например, данная статья может иметь тэги Статьи, Техническая информация и Linux.

Бизнес для зевак

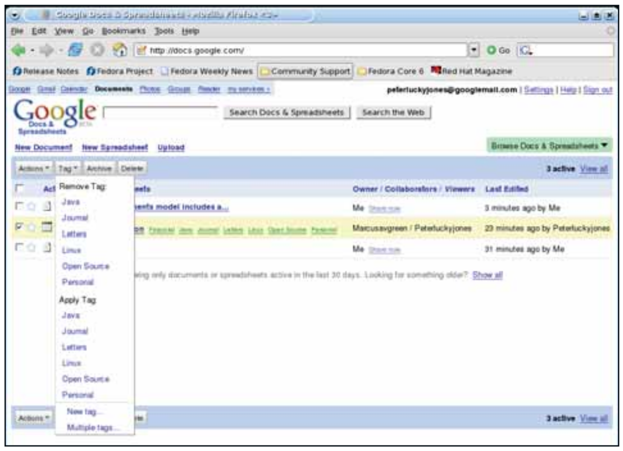
Хотя блокировка всплывающих окон в наших браузерах и защищает нас, пользователей Linux, от онлайн-рекламы, стоит помнить, что Google занимается рекламой, и перед ним стоит цель, чтобы как можно больше глаз увидело его страницы. Подобно интерфейсу 'lo-fi' его офисных инструментов, чисто текстовая реклама Google создана, чтобы быть эффективной, но ненавязчивой. В данный момент инструменты *Docs & Spreadsheets* не содержат никакой рекламы – в отличие от системы Gmail. Возможно, реклама появится в будущем, а может, Google считает, что эти инструменты и сами способны привлечь пользователей, чтоб они и дальше выбирали программы Google. Многие пользователи проходят через учетную запись Gmail, чтобы попасть в *Docs & Spreadsheets*, и пока они находятся в системе Gmail, их глаза упираются в несколько тактичных рекламных объявлений Google, а счетчик впечатлений от Google вновь накручивает обороты.

Google Docs

Интересно, использовал бы это Керуак, чтобы в наши дни написать «*В дороге*» [Д.Керуак – писатель-легенда «спонтанной прозы», – прим. пер.]?

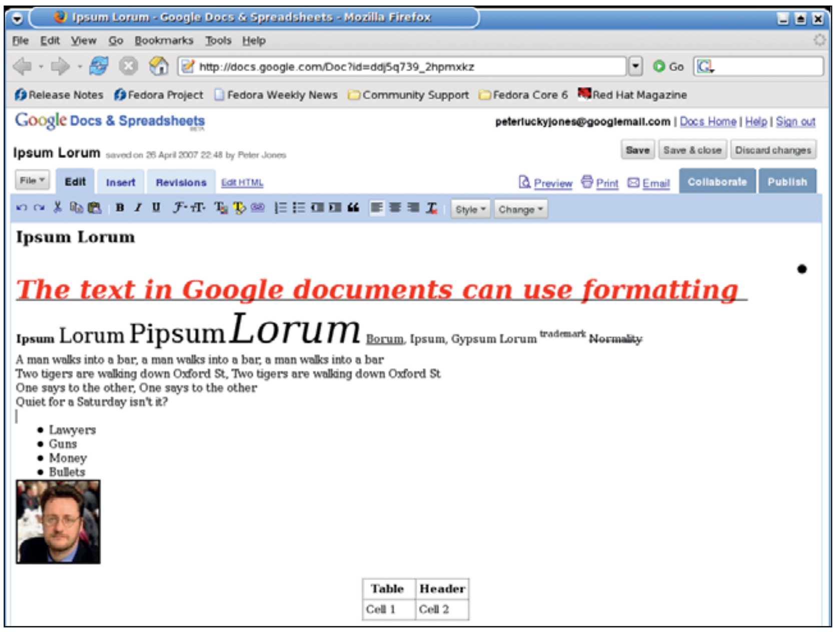
Часть пакета Google под названием *Docs* уходит корнями во *Writely*, старый текстовый редактор для web. Google поглотил создателя *Writely*, Upstartle, весной 2006, и следующей осенью выпустил *Docs* вместе со *Spreadsheets*. Подобно модулю *Spreadsheets*, *Google Docs* – минималистский продукт: он не годится для работы с большими документами или для издательского дела. Ну, по крайней мере, в интерфейсе вы не запутаетесь. Он позволяет просто ввести текст, произвести минимальное форматирование и проверку орфографии и сохранить его, да еще можно вставить таблицы и ссылки. Эта программа включает использование стандартных уровней заголовков – с первого по третий, но не позволяет создавать стили, настраиваемые пользователем.

Немалое преимущество – то, что ваша работа всегда у вас под рукой. Данная статья была создана и отредактирована с помощью *Google Docs*, и доступ к ней был получен с трех разных машин в разных местах и из Linux, и из Windows – в *Google Docs* отсутствует проблема совместимости формата файлов. А еще здесь до смешного просто разделять доступ к документу. Если вы нажмете на Collaborate, ваши друзья и коллеги получат возможность просматривать документ и вносить в него изменения, а вы потом увидите все внесенные ими изменения, до последнего бита, и решите, стоит ли их сохранять. Вам не придется возиться с флэш-картами, дискетами и прочими устройствами для переноса данных. Модуль *Google Docs* ограничивает размер файла до 500 Кб, чего, вероятно, достаточно для большинства людей и большинства целей. Чтобы вы яснее представили себе, сколько это, сообщаем, что документ, содержащий 50 000 слов и несколько рисунков, занимает приблизительно 1000 Кб.



› Использование меню тэгов в модуле Documents.

› **Меню File для модуля Documents.**



Функция проверки орфографии работает отлично. Более ранние версии придерживались стандарта американского английского (US English), но, похоже, это уже исправили. Есть также функция, позволяющая добавлять слова в ваш личный словарь. При запуске проверки орфографии слова подчеркиваются, а если нажать на подчеркнутое слово, появится контекстное меню со списком вариантов правописания или функцией добавления в словарь. Это явно лучший интерфейс, чем отдельная обработка слов, подобная *OpenOffice.org* или *MS Word*.

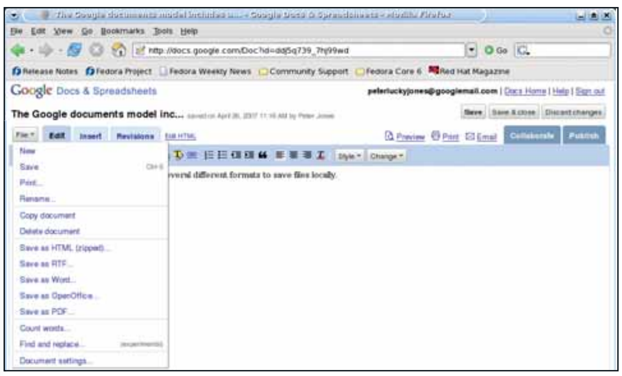
Проверка грамматики отсутствует, но есть функции предпросмотра и печати, и вы можете сохранить свой документ в формате **PDF** или **ODT**-формате *OpenOffice.org*. Будь вы пользователем *Windows MS Office*, одна лишь функция создания файлов в формате **PDF** стала бы отличной причиной помчаться за учетной записью.

Помогите!

На каждой странице *Docs & Spreadsheets* есть ссылка на файл подсказки, но, честно говоря, эта подсказка не слишком хороша. Так, например, подсказка по формулам выдает следующую информацию:

«Google Docs & Spreadsheets поддерживает свыше 200 функций формул (например,. «=sum(A1:A10)»), что позволяет вам импортировать и производить довольно сложные расчеты в своей таблице. Для доступа к общим функциям перейдите на вкладку «Формулы» или нажмите на ссылку «more», чтобы просмотреть полный список поддерживаемых функций и легко вставить их в редактируемую ячейку».

Инструментам *Docs & Spreadsheets* не хватает документации от третьих лиц, и мы подозреваем, что потенциальные авторы жаждут поглядеть, как пойдут дела у продукта, прежде чем приняться за книги.



› **Вот пример форматирования текстового документа.**

Как приступить к работе

Чтобы использовать Docs & Spreadsheets, вам нужно создать учетную запись в Google. Она даст вам доступ к сервисам Google, а прямая ссылка — это http://docs.google.com.

Если вы уже являетесь пользователем Gmail, вам просто надо кликнуть по ссылке на Docs & Spreadsheets в стандартном почтовом интерфейсе. Бесплатный вариант пользователя дает вам 2 ГБ пространства, а для текстовых документов это огромный объем. Если вы его превысите, Google предложит объем 10 ГБ Premier Edition за $50 в год. См. www.google.com/a/help/intl/en/admins/editions_spe.html.

Альтернативы

Вас не привлекают инструменты Google? Познакомьтесь с другими онлайн-решениями.

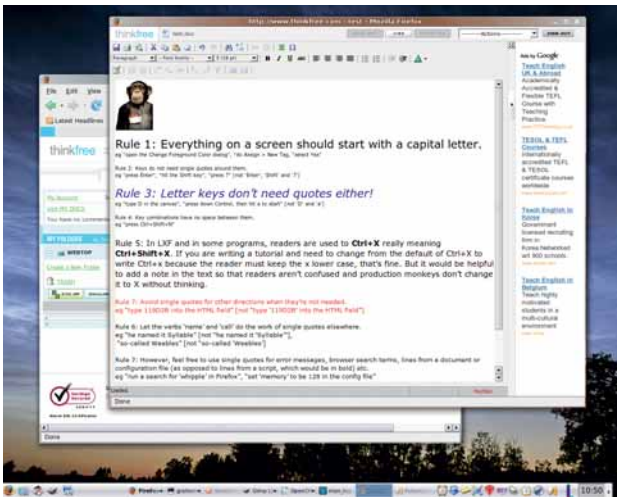
Google – не монополист рынка приложений на базе web: среди его конкурентов – офисный пакет *Zoho* (www.zoho.com) и *ThinkFree* (www.thinkfree.com). Вообще-то, если сравнивать только по функциям, эти продукты еще и круче инструментов Google. В отличие от Google, *Zoho* не жалуется, если вы пытаетесь зайти в программу через *Opera* под Linux или *Konqueror*. Но, к сожалению, и работать нормально в любом из перечисленных браузеров она не будет.

Пакет *Zoho* включает приложение для создания презентаций и множество инструментов помельче, а также текстовый редактор и редактор таблиц (хотя Google недавно объявил о добавлении к своим сервисам программы для создания презентаций на базе web). Интерфейс у модулей *Zoho* намного более лощеный, чем у инструментов Google, и реализует больше функций, которых как-то ждешь от программы для рабочих столов. Модули копируют интерфейс отдельных приложений: например, кнопки текстового редактора напоминают кнопки в *Excel* или *OOo*. Вкладки в редакторе таблиц имеют скошенные края, и в нем есть панель ввода формул. Инструменты *Zoho* немного напоминают компоненты *Office 97*, тогда как инструменты Google выглядят так, словно создавались для текстового интерфейса.

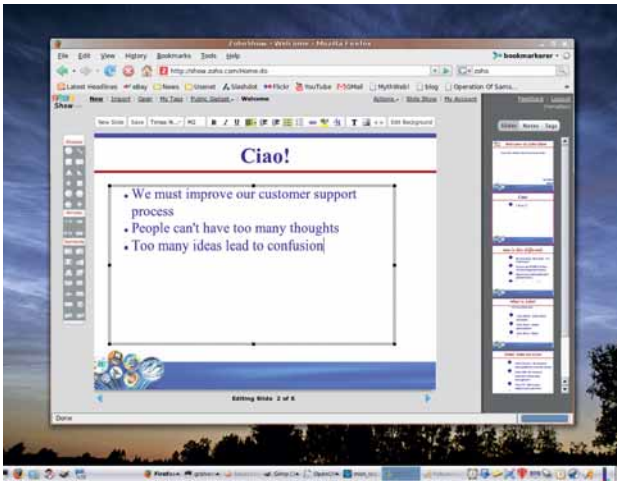
Мы подозреваем, что за красивость инструментов *Zoho* нужно расплачиваться временем, требуемым на их закачку, однако с учетом качества соединения современного Интернета это не должно стать чересчур дорогой ценой. Таблицы *Zoho* обладают функцией автозаполнения, которая работает точно так же, как и подобная функция в *MS Office* и *OOo*. Более важно, что таблицы *Zoho* поддерживают неплохую часть графической функциональности стандартных электронных таблиц. Можете выбрать область данных с помощью мыши и нажать на кнопку создания графика, и запустится соответствующий мастер.

Апплеты – ваши друзья

Инструменты *ThinkFree* интересны тем, что они выходят в двух версиях. Одна – это набор инструментов HTML/Ajax, работающий таким же образом, как *Zoho* и *Google Docs*, а другая называется «Java-апплет». Если вы скрипнули зубами при мысли о Java-апплетах, не торопитесь делать выводы – это не те апплеты, с которыми мучился ваш дедушка. Использование Java позволяет создавать намного более сложные и тонкие приложения и интерфейсы, чем HTML/Ajax. Инструменты *ThinkFree* больше остальных соответствуют интерфейсу оффлайновых приложений, однако цена, которую приходится платить за загрузку кода апплета – это производительность.



› Как видите, пользовательский интерфейс ThinkFree более утончен, чем его эквивалент в Google. Есть и корпоративная версия.



› Тех, кого огорчает отсутствие инструмента для создания презентаций от Google, может заинтересовать модуль *ZohoShow*.

Ни *Zoho*, ни *ThinkFree* даже не пытаются предложить пользователям свободу, присущую свободному ПО или лицензиям GPL или BSD. Google дает доступ к некоторым исходным инструментам в своих системах, но не предлагает ничего такого, чтобы каждый мог настроить дубликат этой системы.

Просто чтоб придраться

Конечно же, во всех упомянутых нами инструментах есть недостатки и ограничения. Но на самом деле, это уже просто придирки. С тем же успехом можно пожаловаться на то, что вместе с купленным телевизором не доставили даму, продающую в фойе кинотеатра мороженое во время антракта, или что размер изображения не достигает 30 футов в высоту. Это, конечно, прискорбно, но пренебрежимо по сравнению с тем, что теперь вы можете развлекаться, рассевшись у себя дома, а для просмотра фильма не надо платить за билет.

Ценность web-приложений будет расти, поскольку они предлагают такое потрясающее удобство, как распределенный и универсальный доступ. Возможно, недостаток лоска у инструментов Google компенсируется связанным с ними брэндом. Для пользователей Linux, довольствующихся *Firefox*, эти инструменты предлагают весьма солидные преимущества. LXF

Мнение от OpenOffice.org

Майкл Микс [Michael Meeks] – глава группы *OpenOffice.org* в Novell. Беспокоит ли его рост популярности инструментов Web 2.0?

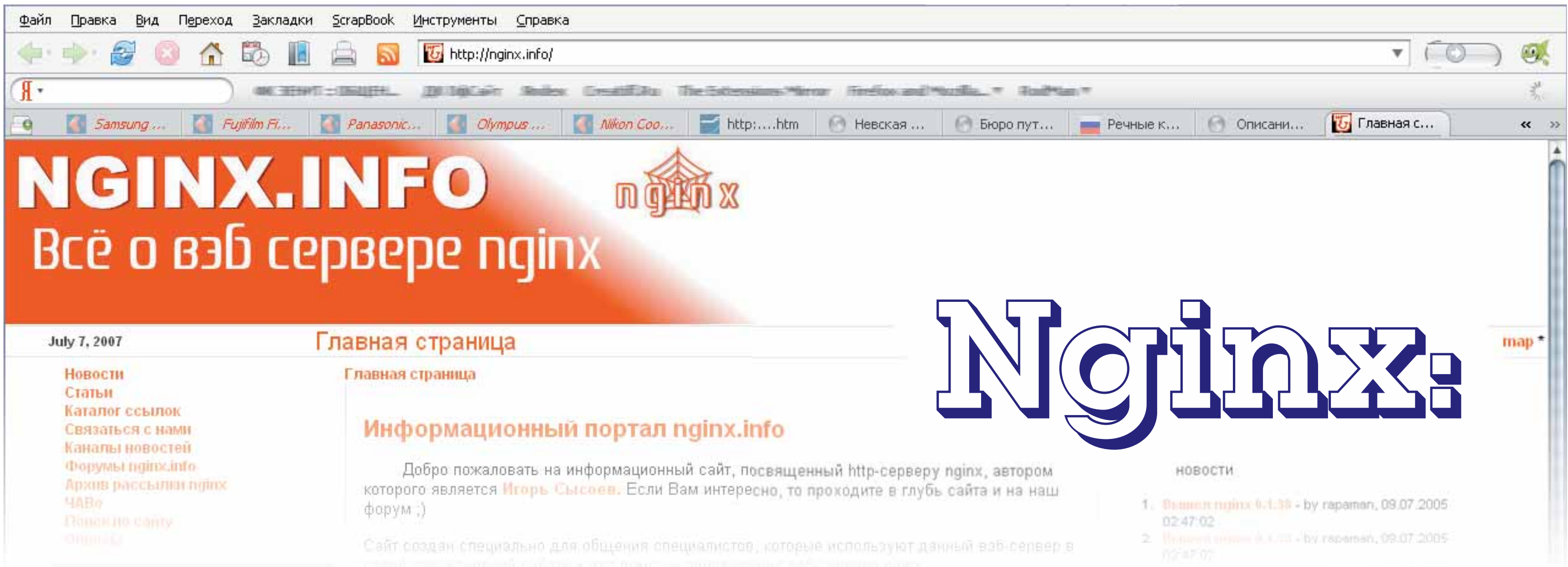


«В то время, как количество и возможности офисных онлайн-приложений Web 2.0 продолжает увеличиваться, остаются как минимум несколько проблем, скрывающихся за фасадом – например, редактирование текста: многие из «текстовых редакторов» – это просто обертка на HTML, что очень сильно затрудняет редактирование мощного WYSIWYG-документа. Сложность представляют и серьезные вычисления, и экранный рендеринг.

При всем при том вполне очевидно, что они весьма привлекательны для множества мелких задач, и что, так же как с wiki, я думаю, эти парни проглотят задачи редактирования/обработки более простых документов. Что интересно, некоторые из них используют за сценой *OpenOffice.org* для конверсии документа, так что на каком-то этапе *OOo* будет усовершенствован благодаря этому. В частности, они должны извлекать пользу из постоянно продолжающейся работы Novell над OpenXML и взаимодействия с форматами Microsoft вообще. Но все же, по-моему, маловероятно, что web-приложения как-то повредят открытому ПО. К тому времени, как они станут популярными, люди уже сами смогут писать собственные web-приложения, что – в настоящий момент – кажется вполне реальным, в отличие от возможности переписать *OOo*.»

Новая серия!

Ультрабыстрый открытый web-сервер – отечественная разработка



Nginx:

ЧАСТЬ 1 Интернет немыслим без web-сервера. 60-70 процентов узлов Сети обслуживаются *Apache*, а как живут оставшиеся 20-30 процентов? **Валерия Комиссарова** знает ответ.



Наш эксперт

Валерия Комиссарова
Эксперт Cnews, дипломированный администратор и разработчик решений на C#, Валерия с интересом следит за развитием Unix и умеет по достоинству оценить существующие для него решения.

Web-сервера бывают разные – получше, похуже или вовсе предназначенные для решения нескольких типов задач. Выбрать web-сервер, полностью соответствующий именно вашим нуждам, просто необходимо: от этого во многом зависит, насколько хорошо – эффективно и удобно для пользователей – будет функционировать ваш web-ресурс. Соответственно, правильный выбор и поддержка web-сервера – известная «головная боль» почти всех администраторов в мире. В заголовок данной статьи вынесено название продукта, который мы будем рассматривать; наверное, однозначно назвать его «мечтой администратора» – некоторое преувеличение. Но, без сомнения, *Nginx* – неплохой выбор для попытки приблизиться к главному желанию сисадмина почти любого ресурса: стабильно и эффективно работающему web-серверу.

Кое-что о Nginx

Прежде чем говорить об этой разработке, давайте разберемся, что же такое *Nginx* и какова его история.

Nginx – HTTP-сервер (и одновременно – почтовый прокси-сервер, но об этом позже). Сразу же уясните для себя важную деталь: *Nginx* не является стандартным web-сервером в том смысле, в каком большинство читателей могут его воспринять, т.е. *Nginx* не может служить функциональной заменой, например, *Apache* или IIS. В сущности, *Apache* и *Nginx* находятся «по разные стороны баррикад». Здесь уместно вспомнить модель «front-end/back-end»: соответственно, *Apache* будет относиться к back-end, а *Nginx* – к front-end. Данные продукты предназначены для выполнения задач различных типов – это очень важно.

Почему мы говорим исключительно об *Apache*? Да потому, что *Nginx* работает только под Unix/Linux системами.

Теперь об истории проекта. *Nginx* начал разрабатываться Игорем Сысоевым (одним из администраторов Rambler'а) весной 2002 года. Проект стал использоваться на различных серверах (в частности, в том же Rambler'е) задолго до официального релиза, состоявшегося 4 октября 2004 года, и до своего обнародования *Nginx* уже прошел серьезное тестирование. По данным, собранным и организованным Алексеем Тутубалиным, в марте 2006 года щелчок по «Черному квадрату» (http://www.rukv.ru) приводил к отклику *Nginx* в 9,8%, а в марте 2007 – уже в 22,7% случаев (речь идет о виртуальных серверах). Помимо Rambler'а, стоит упомянуть о Mamba и Peterhost, на чьих серверах также функционирует *Nginx*.

Когда пригодится Nginx?

В чем преимущества использования модели «front-end/back-end»? Основное – значительно бóльшая эффективность работы, чем в других – стандартных и привычных – схемах использования того же *Apache*; но отнюдь не во всех.

Прежде всего необходимо учесть, что использование обсуждаемой модели будет уместно только применительно к серверам с высокой нагрузкой, оправдывающей использование подобного рода средств. Под «высокой нагрузкой» следует понимать количество http-запросов, превышающее 10–12 в секунду. Соответственно, если такой нагрузки на сервер нет, то лучше не строить работу сервера по принципу «front-end/back-end»; это решение может привести и к некоторому замедлению работы (хотя и довольно незначительному): ведь *Nginx* – дополнительный «пропускной пункт» на пути к back-end'у и к увеличению потребления памяти и т.д. Не стоит использовать *Nginx* там, где в нем нет нужды, чтобы не винить в неудобствах и неувязках продукт и его разработчика.

Что умеет Nginx?

Итак, какие возможности предоставляет данный продукт? *Nginx* – это и HTTP-сервер, и почтовый прокси-сервер. Поэтому рассматриваемые возможности оправданно разбить на категории согласно двум функциональным направлениям *Nginx*. Начнем с HTTP-сервера, в числе функций которого:

» **Обслуживание статических запросов**. Конкретнее, это «передача» пользователю статических html-страниц, графических изображений и прочего контента. Второе – обслуживание запросов на индексные файлы. Данная функциональность реализуется модулем ngx_http_index_module: он обслуживает запросы, оканчивающихся слэшем ("/").

мечта сисадмина



» **Автоматическое создание списка файлов**; это задача модуля ngx_http_autoindex_module, который выполняет автоматическое создание листинга каталога. Запрос попадает к модулю в том случае, если у компонента ngx_http_index_module возникли проблемы с поиском индексного файла.

» **Ускоренное проксирование без кэширования**, выполняемое модулем ngx_http_proxy_module. Он обладает большим количеством директив, с помощью которых можно настроить параметры процесса. В следующий раз мы рассмотрим этот вопрос более подробно.

Особого внимания заслуживают функции/модули, касающиеся обеспечения отказоустойчивости виртуального сервера и распределения нагрузки. Данная функциональность связана с модулем ngx_http_upstream.

Предусмотрена еще и поддержка SSL.

Что же касается «почтовой» функциональности, то здесь имеются возможности для нормальной работы с IMAP, POP3, SMTP, а также SSL. Среди поддерживаемых методов аутентификации: LOGIN для IMAP, USER/PASS, APOP, AUTH LOGIN PLAIN CRAM-MD5 у POP3, и AUTH LOGIN PLAIN CRAM-MD5 у SMTP.

Чего *Nginx* «не умеет» и вряд ли будет «уметь»? Судя по «настрою» разработчика, о **.htaccess** придется забыть. Не стоит надеяться и на поддержку CGI.

Что осталось добавить? Возможность обновления исполняемого файла *Nginx* и его настроек без остановки процесса обслуживания клиентов; высокая модульность. Среди приятных «мелочей» – быстрая ротация журналов, ведение отладочного журнала, и перенаправление ошибок (например, 404).

Почему Nginx так быстро работает?

А теперь давайте посмотрим, почему *Nginx* (как и сама модель «front-end/back-end») работает быстро и эффективно (*Nginx* действительно быстр, это не «слухи»). Можно поставить вопрос и так: почему разделение обязанностей между *Apache*/*Nginx* обеспечивает большее быстродействие, чем в схеме работы без последнего?

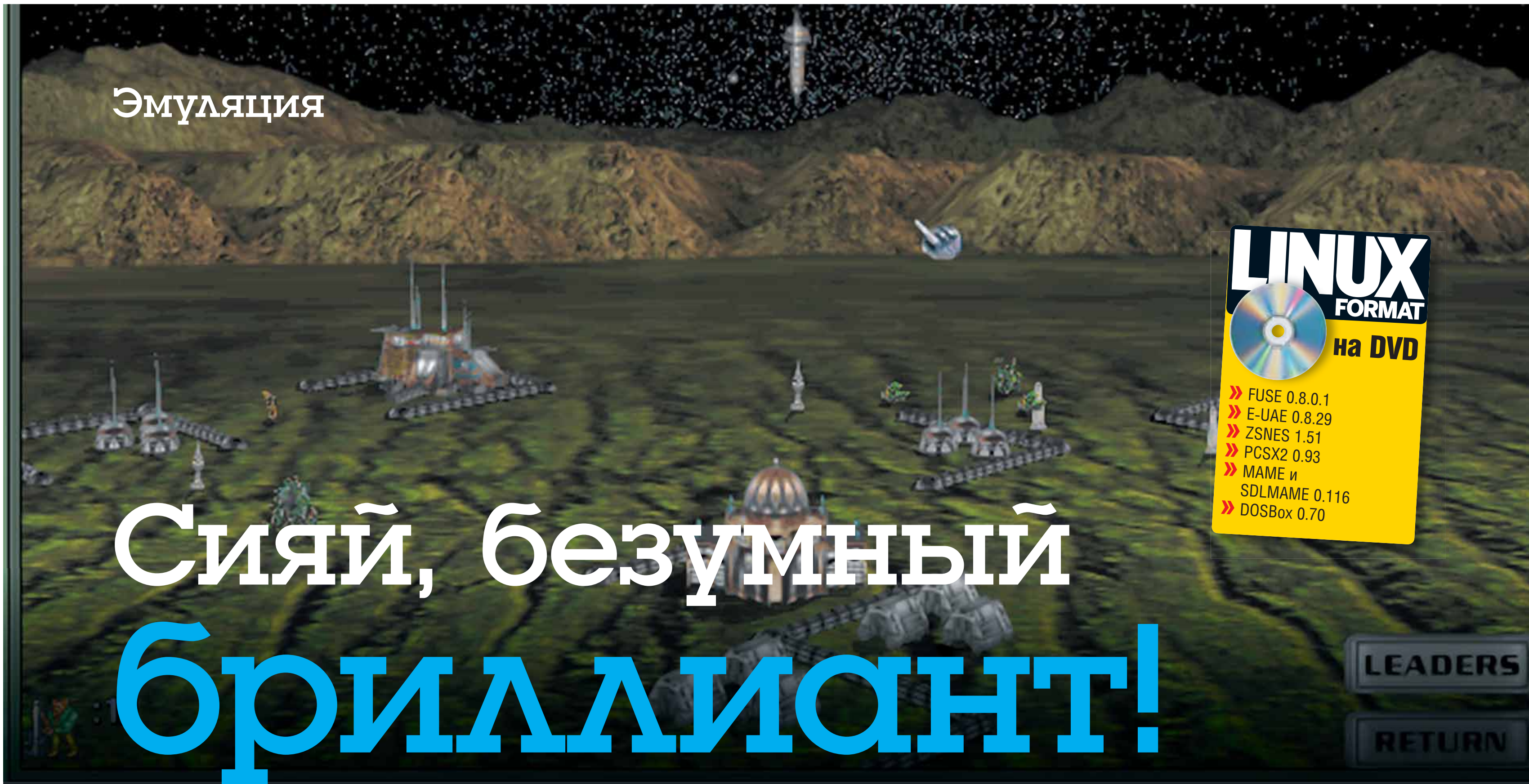
Почему – если сравнивать работу *Apache* без *Nginx* и в связке с *Nginx* в ситуации со статическим контентом – второй вариант быстрее (иногда это заметно больше, иногда меньше, но чувствуется всегда)? Рассмотрим, что такое модель prefork(ed), используемая в *Apache 1.3*. Мы имеем один главный процесс, который при получении входящих запросов создает требуемые дочерние процессы с помощью fork (). Такой подход традиционен, и в то же время является одним из худших по производительности/эффективности. Почему? Происходит серьезный перерасход системных ресурсов (думаю, не нужно объяснять сущность работы fork()). И бороться с тем, что множество дочерних процессов (число которых, понятное дело, растет с числом запросов) стремительно поглощает системную память, негативно влияет на производительность и в разы снижает удобство работы с ресурсом, очень трудно. Что же нам предложит *Nginx*? FSM! Аббревиатура расшифровывается как Finite State Machine – автоматы с конечным числом состояний, иначе – КА (конечные автоматы). Несмотря на ряд ограничений, в ситуации с *Nginx* FSM полностью себя оправдывает. FSM – довольно сложная вычислительная модель; ей посвящен не один толстый книжный том; ну, а мы просто посмотрим, что еще используется в отношении скорости в *Nginx*, и к какому результату, заметно отличающемуся от prefork(ed)-модели, это все приводит.

Nginx использует kqueue (на системах Free/Net/Open BSD и Mac OS X) – механизм оповещения определенного процесса о конкретных событиях, произошедших в ядре ОС. Применение kqueue, в частности, позволяет избавиться от большого количества «лишних» вызовов некоторых функций. Также используются epoll и rtsig (в ОС Linux), sendfile (для Linux, FreeBSD, Solaris). На примере kqueue очевидно направление этих функций. Главное – результат: радикальное уменьшение количества «съедаемой» памяти, увеличение быстродействия и т.д., со всеми вытекающими последствиями.

Эффективность модели «front-end/back-end» напрямую следует из осуществления значительного снижения воздействия негативных сторон prefork(ed) *Apache*. Разделение труда между front-end и back-end дает радикальное уменьшение числа переключений контекста и количества потребляемой памяти.

На самом деле возможностей у *Nginx* намного больше, чем здесь описано. Говорить о них можно если и не бесконечно, то достаточно долго. Но задача данной статьи – рассказать, насколько многообразен и качественен этот web-сервер; детальному рассмотрению процесса установки и настройки *Nginx* будет посвящена следующая статья. LXF

» **Через месяц** Мы установим Nginx и научимся настраивать его в ряде типовых ситуаций.



LINUX FORMAT на DVD
» FUSE 0.8.0.1
» E-UAE 0.8.29
» ZSNES 1.51
» PCSX2 0.93
» MAME и SDLMAME 0.116
» DOSBox 0.70

Сияй, безумный бриллиант!

В Linux работают десятки тысяч игр. **Александр Супрунов** проложит вам путь к ним.

Сквозь время

За последние десятилетия в мире создавалось много уникальных компьютерных систем. В туманном Альбионе за домашним компьютером Клайва Синклера выстраивались огромные очереди жаждущих заполучить в дом электронного помощника стоимостью менее 100 фунтов (что в 1982 году, как и сейчас, примерно равнялось $200), имеющего цветное изображение и ввод команд BASIC одним нажатием клавиши. Покупателей не пугали ни низкое качество сборки компьютеров, ни ошибки, обнаруженные в уже проданных экземплярах, ни даже недоделанное ПЗУ этой машины. И их можно было понять: за 99,95 фунтов купить полноценный цветной чудо-компьютер, в то время как другие модели стоили от 400 – чем не находка? За покупкой, правда, надо было идти не в компьютерный магазин, а в книжный.

А толпы фанатов ***Apple II*** (представлявшей собой уже готовое устройство, а не набор ***Apple I*** типа «собери-сам») боготворили свою машину, созданную двумя Стивами (Возняком и Джобсом) – и было за что. Во времена, когда домашние компьютеры делали первые шаги, ***Apple II*** позволяла выполнять даже такие фантастические вещи, как самостоятельную оцифровку звука.

Чуть позднее (в 1984 году), первые обладатели легендарного Макинтоша гордились графическим интерфейсом этой машины – даже несмотря на то, что экран, встроенный в компьютер, был совсем небольшим (конечно, по сравнению с диагональю современных мониторов) и черно-белым. Впрочем, и внутри Apple разработкой ***Macintosh*** занимались элитные сотрудники (большей частью – пришедшие из проекта Lisa); тех же, кто продолжал работать над приносящей основную прибыль ***Apple II***, Джобс считал неудачниками (но это уже другая история). И, конечно, первая игра для Макинтош – *Alice* – была по-настоящему хороша.

Дополнительная информация

Конфигурация тестовой машины:

Процессор:	AMD Sempron 3000+ (1600 МГц)
ОЗУ:	512 МБ DDR PC-3200
Видеокарта:	128 МБ GeForce 5200

В том же 1984 году в Нью-Йорке был продемонстрирован прототип машины, повергший в шок невероятной мощью даже видавших виды журналистов: сверхбыстрая графика, потрясающий звук, многозадачная операционная система, манипулятор «мышь» и 4096 цветов на экране.

Конечно же, это была ***Amiga***. Мультимедиа в 1984 году! У вас и сегодня может отвалиться челюсть при виде живого видео со стереозвуком, легко демонстрируемого с 880-килобайтной дискетки на полный экран машины с процессором в 7 МГц. Плавно и без тормозов.

Неугомонный Синклер представил общественности новый компьютер ***Sinclair QL*** – призванный затмить 32-битной мощью медленные ***IBM PC***. Именно его, замечу, в свое время приобрел Линус, соблазнившись теми самыми 32 битами. Позже он купил себе IBM PC и подарил нам Linux.

И это было только начало.

Сегодня, в отличие от тех времен, когда мир полнился гениальными идеями о создании новых домашних компьютеров, все значительно скучнее: максимум, из чего можно выбирать – это x86 или... или... x86. Последний оплот альтернативности для домашнего пользователя – ***Mac*** – пал под натиском Intel-совместимости в 2005 году.

Ну, мы расстраиваться не будем. Во-первых, Linux – сам по себе вещь весьма увлекательная, конструктор для взрослых: можно вечно собирать и разбирать. А во-вторых, все те замечательные машины, что некогда имели армии фанатов, теперь доступны посредством эмуляции. Разве могли бы вы мечтать о двадцати, а то и тридцати уникальных компьютерных системах прямо в вашей комнате?

Речь идет об эмуляторах – специальных программах, виртуально воссоздающих легендарные компьютерные системы. Превратить ваш мега-супер-компьютер в старый добрый ***ZX Spectrum*** или ***Commodore 64*** – да нет проблем! Только не рвите на себе волосы – процесс обратим, в отличие от лысины.

Где вы, бравые пилоты *Elite* с BBC? У вас есть еще один шанс испытать судьбу, сражаясь у далеких звезд, заправляя под завязку топливные баки от раскаленных светил или кроша астероиды в погоне за удачей.

И не только пилоты. Миллионы людей, переживавших приключения юной Занции в Кирандии, Джеймса Понда или Сквола – все мы снова можем вернуться в этот чудесный мир благодаря эмуляции.

Все, что для этого нужно – Linux и эмуляторы.

ZX Spectrum

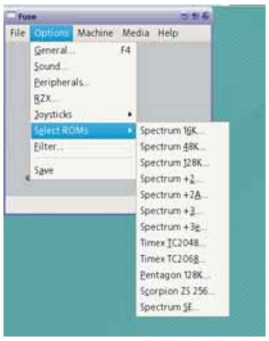
Эмулятор: FUSE **Сайт:** fuse-emulator.sourceforge.net

В мире существует чрезвычайно большое количество фанов ***ZX Spectrum***. Несмотря на весьма скромные, я бы даже сказал, чрезмерно ограниченные технические параметры компьютера, игры с этой платформы по-прежнему вызывают живой интерес, бесконечно скачиваются из сети Интернет и покупаются на CD-дисках. Все дело в том, что они, в отличие от современных высокобюджетных коммерческих блокбастеров, имеют совершенно непередаваемую атмосферу.

Положа руку на сердце, скажите, неужели в играх с абсолютной свободой действий в пределах 48 Кб, типа *Elite*, вам на самом деле будет важна графическая составляющая? Заметьте, когда к вашему потрепанному кораблю с трех направлений подбираются сговорившиеся космические пираты, нещадно поливающие обшивку смертоносными лучами, а защитные экраны сдаются один за другим, в голове бьется только одна мысль: «Ох, дотянуть бы до станции типа Кореолис».

Или, например, где вы встретите игру, в которой путешественник во времени (Уинтроп) попав из XX века в XXV, не желает возвращаться назад, даже несмотря на то, что в его теле в 20 веке сейчас находится человек из 25 столетия. Ваша задача – в течении 30 минут успеть убедить путешественника, что мир будущего не столь уж прекрасен, как видится ему.

Да! Когда в арсенале лишь крохотный объем памяти домашнего компьютера, программисты способны на чудеса при создании новых игр.

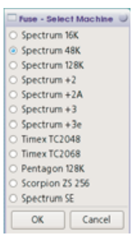


Ну, что же, возможно я вас убедил попробовать пару хитов из звездной копилки ***ZX Spectrum***. А возможно, вас убеждать и не надо было. Так или иначе, для того, чтобы превратить ваш современный компьютер в медленного неповоротливого монстра нам понадобиться хороший эмулятор. На эту роль больше всего, на мой взгляд, подходит *Free Unix Spectrum Emulator – FUSE*. Он предоставляет такой же комфорт работы с воссозданной ретросистемой, как и лучшие представители эмуляторов из мира Windows и способен работать как одна из следующих моделей: ***Spectrum 16K/48K/128K/+2/+2A/+3/+3e/SE***, ***Timex TC2048/TC2068***, ***Pentagon 128*** и ***Scorpion ZS 256***. *FUSE* поддерживает загрузку запакованных файлов, воспроизводит звук через OSS (Open Sound System) и эмулирует основные органы управления: джойстики Kempston, Sinaclair и Cursor.

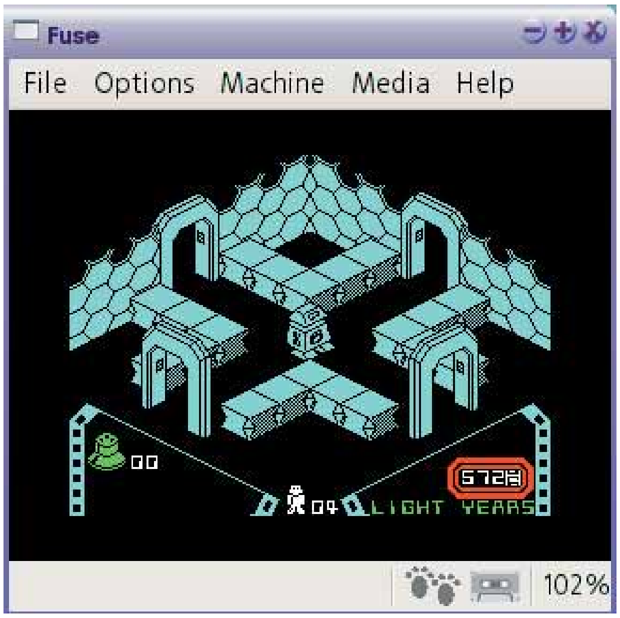


Для сборки *FUSE* из исходников потребуется несколько библиотек: *libgcrypt*, *libpng*, *libxml2* и *zlib*, которые, вероятно, уже есть в вашей системе. На этапе настройки (**./configure**) можно включить поддержку фреймбуфера, SDL, SVGAlib или GTK.

FUSE легко пользоваться, в чем можно убедиться, взглянув на снимки экрана. Основные настройки расположились на вкладках File, Options, Machine, Media, Help. Обратите внимание, что любому пункту меню можно назначить горячие клавиши, просто нажав их в тот момент, когда курсор находится на интересующем пункте.



Вкладка Help содержит картинку с раскладкой клавиатуры ***ZX Spectrum***. Не забывайте, что в этом компьютере набор команд производится не посимвольно, а нажатием одной или нескольких клавиш. Например, чтобы напечатать слово LOAD достаточно нажать клавишу J.



› ***Alien8*** **– звезды, космос, роботы – что может быть увлекательнее.**

На вкладке File доступна одна из самых используемых опций: Open. Как следует из названия, она позволяет открыть образ игры в эмулируемой машине. Здесь же вы сможете сохранить снимок экрана или сделать снимок памяти, который позволит вам начать игру в следующий раз с того же места.

Во вкладке Options скомпонованы основные настройки – полное раздолье для любителей все подстраивать под себя. Переназначайте клавиши, изменяйте ПЗУ машины, скорость эмулируемого компьютера (разгоните ***ZX Spectrum***!) и многое другое.

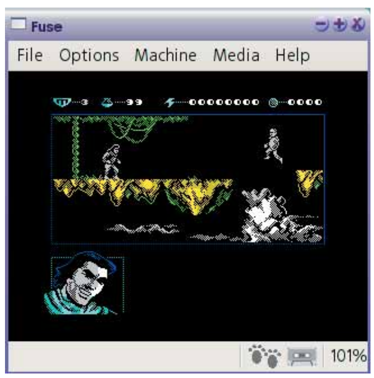
Во вкладке Machine можете изменить модель. Если вы захотите запустить модели ***+3*** и ***+3e***, то вам понадобиться еще одна специфическая библиотека – *lib765*, которая доступна по адресу: http://www.seasip.demon.co.uk/Unix/LibDsk/.

Общая инструкция по работе с эмулятором может выглядеть так: во вкладке File выберите пункт Open и щелкните на интересующем вас образе (поддерживаются .Z80 – снимки памяти или .TAP – образы аудиокассет). Затем играйте и наслаждайтесь!

»







Совет

Альтернативой FUSE может стать запуск одного из Windows-эмуляторов (рекомендуем отечественный *ZXMak* – http://zxmak.narod.ru/) в WINE. Если у вас достаточно мощный процессор, можете запустить в *DOSBox* классический эмулятор Шалаева.



» Commodore AMIGA

Эмулятор: E-UAE **Сайт:** www.rcdrummond.net/uae

Прототип компьютера ***Amiga*** был впервые продемонстрирован на выставке Consumer Electronics Show 4 января 1984 года в Нью-Йорке, в Линкольн-центре. В качестве демонстрации по экрану летал белый шар в красную клетку, с грохотом ударяясь о края экрана («Bouncing Ball»). Мнение присутствующих, после технологического шока, свелось к тому, что наступила новая компьютерная эра.

В продажу уникальный компьютер попал в 1985 году и благодаря фантастической мощи использовался многими исключительно для игр. В его основе лежал центральный процессор Motorola 68000 7 МГц, но это была исключительно быстрая машина, обладающая многозадачной операционной системой AmigaOS с революционной библиотекой Intuition, ориентированной на работу с манипулятором типа «мышь». Секрет этой мощи крылся в уникальных специализированных чипах Agnus, Portia и Daphne, использовавшихся для вывода графики, звука, управления портами и экраном. Они позволили в 1985 году легко выполнять на несравненно более дешевой *Commodore AMIGA* ту же работу, что и на компьютерах стоимостью 50 000 долларов. AMIGA стала активно использоваться для создания видео-эффектов в ультрамощных блокбастерах *Терминатор 2*, *Джеймс Бонд*, *Вавилон 5* и других. Причем в *Вавилоне 5*, благодаря мощи объединенных компьютеров ***Amiga***, впервые удалось на экране отобразить более 100 космических кораблей – на тот момент совершенно невиданное зрелище!

А теперь немного сухих технических данных.

Commodore AMIGA способна отображать до 4096 цветов (режим HAM – Hold and Modify), демонстрировать живое видео на полный экран, проигрывать многоканальный звук (в формате MOD), выводить одновременно на экран окна в разном разрешении и просто убивать наповал многозадачностью, ориентированной на применение в настольных системах. На этом компьютере очень много высококлассных красивых игр (достаточно назвать *Lion Heart*, *Brian The Lion*, *Stardust*) и уникальных программ (*Deluxe Paint*, *Lightwave 3D*, *Directory Opus*), которые постоянно пытаются воссоздать на современных системах, но пока безрезультатно (пример: замена *Directory Opus 4* – файловый менеджер Gentoo).

Все эти скины для проигрывателей, перетаскивание контента на иконку программы для запуска и многое другое уже давным-давно было на ***Amiga***. Аудио воспроизводилось без участия центрального процессора, видео – аналогично, поэтому пользователи ***Amiga*** всегда работали под фоновую музыку. Всего при создании ***Amiga*** было реализовано более 50 ноу-хау, позже запатентованных.

Все модели ***Amiga*** можно глобально разделить на два поколения машин. Первое – на чипсетах OCS (Original Custom Chip)/ ECS (Extended Custom Chip) – это ***A1000/2000/3000/500/500+/600***. Второе – на революционном чипсете AGA (ранее AA), где были сняты ограничения на количество цветов, серьезно увеличена скорость – ***A4000/A1200/A5000*** и т.д.



Модели ***A1000/2000/3000/4000*** были профессиональными машинами в корпусах настольного типа с возможностью расширения. Модели ***A500/500+/600/1200*** умещались в клавиатуре и предоставляли ту же мощь за значительно меньшие деньги. Присутствовали также и возможности расширения, хотя и не в таком объеме, как у профессиональных вариантов.

Большое распространение на платформе ***Amiga*** получили акселераторы, увеличивающие мощность домашней ***Amiga*** в 5, 10 и даже 50 (!) раз. В нашей стране хитом был Blizard 68030 50 МГц благодаря невысокой цене и отличному быстродействию.

Ввиду высокой сложности платформы, на сегодня существует всего два основных эмулятора ***Commodore AMIGA***: *xfellow* и *UAE*.

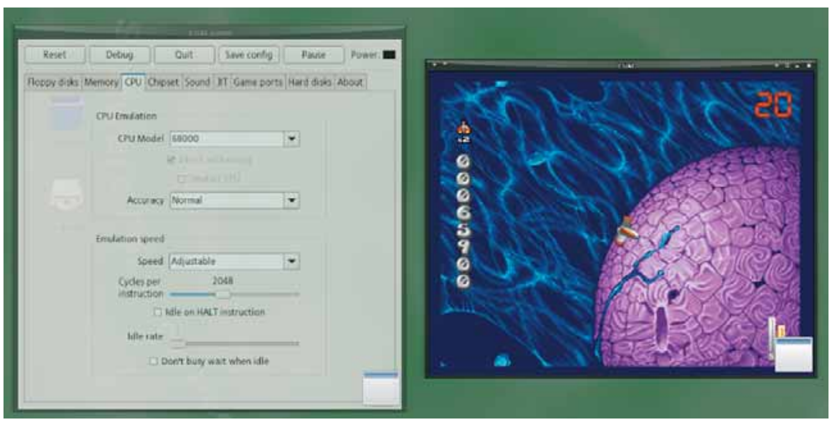
Я же рекомендую *E-UAE,* основанный на *UAE*. Он имеет совершенно фантастическую совместимость с легендарной машиной и способен изменять настройки в процессе работы. Эмуляция потребует образа ПЗУ ***Amiga*** – *kickstart*. Для игр, созданных для первых чипсетов OCS или ECS, часто достаточно *kickstart v1.3*, для ECS-основанных игр – *kickstart v2.0*, а для AGA-игр – *kickstart v3.0*. Следует отметить, что зачастую ECS и AGA-версии одной игры существенно различаются – дабы показать пользователю все прелести AGA-чипсета.

Самые первые игры для ***Amiga*** – в основном, примитивные порты с ATARI ST не раскрывали возможностей супермашины. А вот с конца 80-х – начала 90-х графика просто поражает: посмотрите на *SuperStardust* и *Kid Chaos*.

Хочу добавить, что я был счастливым обладателем ***Commodore AMIGA 600HD*** (выпуска 1992 года) – чему безмерно рад, так как сей факт внес в мою жизнь громадное количество положительных эмоций.

E-UAE обладает великолепным графическим интерфейсом, поэтому настроить его не составит труда. Только обратите внимание: эмулятор стоит на паузе, и для начала его работы надо кнопку паузы отжать.

Привожу несколько стандартных конфигураций для настройки популярных машин.



Конфигурация 1 (AMIGA 600):

- **Chip Mem:** 2 МБ (расширенная на 1 МБ для уменьшения количества подгрузок)
- **Slow Mem:** нет
- **Fast Mem:** нет
- **Kickstart ROM:** Kickstart v.2.0 (версия 37175)
- **CPU Model:** 68000 (Accuracy: normal / Speed: Approximate 7 Mhz)
- **Chipset Model:** ECS
- **Sprite Collisions:** Full

Пренебрежение этим пунктом может повысить скорость эмуляции, но в таких играх, как *Deluxe Galaga*, вы обнаружите, что пули перестали причинять врагам повреждения, а призы не подбираются.

- Обратите внимание, что включение галочки Immediate Blits может

нарушить некоторые эффекты в играх (например, в *Elfmania* перестают низвергаться водопады).

» **Sound:** Stereo (Accurate, 16 bit)

В настоящей ***Amiga*** звук 8-битный. На более мощных моделях с AGA-чипсетом пользователи стали использовать драйвер AHI, позволяющий получить 12-битный звук, но поверьте, и при 8 битах «амижный» стереозвук великолепен и просто завораживает своим качеством, приятным тембром и своеобразной «хрипотцой» на высоких частотах. Звук на ***Amiga*** – как вино с хорошей выдержкой: пьянит с первого глотка.

» **Управление:** выбирайте – Numeric pad или джойстик (если есть).

Конфигурация 2 (AMIGA 1200):

» **Chip Mem:** 4 МБ (по умолчанию продавалась с 2 МБ на борту, но добавление еще 2 МБ увеличивало скорость A1200 в 2 раза!)

» **Slow Mem:** нет

» **Fast Mem:** 8

» **Kickstart ROM:** Kickstart v3.0 (A1200) rev 39.106 (512k)

» **CPU Model:** 68020

» **Chipset Model:** AGA

» **Sprite Collisions:** Full

» **Sound:** Stereo (Accurate, 16 bit)

Почти все игры на ***Amiga*** рассчитаны на использование джойстика (без него на настоящей Amiga играть не получится). Я первое время использовал джойстик от ***ZX Spectrum***, но спустя некоторое время переделал геймпад от ***SEGA Megadrive***.

Если вы выбрали Numeric pad, клавишами управления будут: 8 – вверх, 2 – вниз, 4 – влево, 6 – вправо, 5 – огонь. Дополнительными клавишами могут быть: Space (использование бонусов в *Brian the Lion*), 0 (Ins), Shift (совместно со стрелками влево-вправо – смена оружия) и другие. Общий принцип управления таков:

» Без нажатой клавиши «огонь» – движение и прыжки.

» Нажатие клавиши «огонь» и любой другой – новое действие. Могу сказать, что такое управление действительно удобно (на джойстике) – оценить правоту слов можно, поиграв в знаменитую «амижную» драку Body Blows.

Качество графики на ***Amiga*** изумляет и завораживает. Я использовал и использую очень много различных игровых систем, но если и в 2007 году я в тридцатый раз прохожу *Stardust* или *Banshee*, то, вероятно, в этом действительно что-то есть.

SUPER NINTENDO

Эмулятор: ZSNES **Сайт:** www.zsnes.com

Если бы Фусадзиро Ямаучи {Fusajiro Yamauchi] не играл в карты, не было бы ***Super Nintendo***.

Забыл добавить – «в 1889 году». А если уж быть совсем точным – не начни он продавать игральные карты, вырезанные из коры тутового дерева, не основал бы и компанию Nintendo Koppai.

Как вы знаете, эволюция сделала из обезьяны человека. Так и компания по продаже карт превратилась в мегамонстра электронных развлечений Nintendo Co. Ltd. Нас же интересует продукция этой компании, а точнее, ***SNES***.

Игровая приставка ***Super Nintendo*** была задумана как убийца ***SEGA Megadrive/Genesis***. И убийца был создан – правда, довольно странный.

SNES поступила в продажу, когда MD существовала уже около трех лет – для приставки весьма солидный возраст. Чем же должна была блеснуть новая система, чтобы не осрамиться перед старичком? Безусловно, 256-цветной графикой и потрясающим звуком.

К сожалению, ***SNES*** обладала крайне слабым центральным процессором – всего 3,58 МГц. Из-за такого досадного ограничения на приставке очень много красивых, задумчивых игр, а вот боевики, где врагов укладывают пачками в вагоны и на экране сверкают десятки взрывов, увы, можно по пальцам пересчитать.

Из за чего захочется поиграть под Linux в ***SNES***-овские игры? О-о-о! Один *SuperMetroid* чего стоит – превосходная научно-фантастическая похлебка с путешествиями на чужие планеты. А ведь есть еще *Final Fantasy 5*, *Chrono Trigger*, *Flinstones*, *Final Fight 2* и другие шедевры.

Нам здорово повезло: для Linux существует лучший эмулятор этой консоли – *ZSNES*. *ZSNES* – открытый эмулятор *Super Nintendo Entertainment System*, написанный на ассемблере, C и C++. Его создает достойная команда программистов – zsKnight, _Demo_, pagefault, Nach и Iheubt. Эмулятор великолепен: он справляется почти со всеми играми для ***SNES***, включая уникальную *Star Ocean*, и уверенно движется к 100% совместимости.

Пока еще не доведена до совершенства эмуляция специальных чипов, встраиваемых в некоторые картриджи, чтобы удивить игроков видеоэффектами и объемной графикой. Но даже такой известный чип, как SuperFX, воспроизведен на 90%. Это означает, что вероятность найти «ром» с игрой, который был бы не по зубам *ZSNES*, крайне мала.

Если вы все же жаждете увидеть это своими глазами – в качестве примера приведу *Batman returns*.

Но, разумеется, игры, созданные для использования совместно со спецчипами, более требовательны к ресурсам вашего компьютера. В среднем, необходима машина с процессором частотой от 500 МГц (при использовании X) или 266 МГц (без X), 64 МБ ОЗУ. Также очень желательна 3D-видеокарта.

ZSNES может использовать (по вашему желанию) билинейную фильтрацию, позволяя насладиться гладкой (без пикселизации) картинкой и фантастической плавностью.

Эмулятор обладает превосходным интуитивно понятным графическим интерфейсом и может работать как в окне, так и в полноэкранном режиме. Из интересных дополнений отмечу возможность наполнить задний план эффектами вроде снега или кругов на воде.

Иными словами, несколько тысяч превосходных игр ждут не дождутся возможности украсить экран вашей Linux-машины и украсть у вас пару тысяч часов свободного времени.



Совет

Длиннющие ролики в RPG-играх можно быстро прокрутить увеличив скорость эмуляции удержанием клавиши “~”.

Для сборки *ZSNES* из исходных кодов понадобится установить ассемблер nasm.

После запуска *ZSNES* вы увидите окно с надписью, приглашающей нажать клавишу Space. Далее выберите пункт GAME –> LOAD и в открывшемся диалоге выбора файлов выделите «ром» с игрой. Я запустил великолепный платформер *The Flinstones*.

После запуска игры первым делом подстройте эмулятор. Нажмите ESCAPE и выберите меню CONFIG. Здесь нас больше всего интересует пункт VIDEO. Он открывает окно с двумя вкладками – MODES и FILTERS. В MODES можно выбрать разрешение окна и режим (полноэкранный / оконный). В FILTERS для борьбы с пикселями отметьте пункт BILINEAR FILTER – это превосходный вариант получить изображение игры, схожее с тем, что можно наблюдать на телевизоре с подключенной настоящей ***SNES***.

Если вам покажется, что музыкальное сопровождение в играх звучит недостаточно хорошо, это можно легко исправить в меню CONFIG –> SOUND: изменяйте частоту дискретизации [sampling rate] вплоть до 48 000 Гц, добавляйте surround – в общем, делайте, что душа пожелает.

Стандартное управление (по умолчанию это курсорные клавиши и Z/X/S/A/D/C) изменяется в меню CONFIG –> INPUT – настройте клавиши «под себя», а лучше подключите геймпад!

Дополнительные клавиши, используемые эмулятором:

F1 – быстрое меню к основным опциям.
F2 – сохранить игру.
F3 – выбрать другой слот для сохранения (от 0 до 9).
F4 – загрузить ранее сохраненную игру.
P – пауза.

И вот теперь, после *Putty Squad*, *Rock'n'Roll Racing*, *Final Fantasy*, *Alien 3*, *Prince of Persia*, *Tetris Attack*… у вас все еще хватает наглости говорить, что под Linux нечем заняться серьезному человеку?

Ну что же, тогда держитесь!

Совет

Для корректной работы Final Fantasy X необходимо применить fix нажатием клавиши F9.

SONY PLAYSTATION 2

Эмулятор: PCSX2 **Сайт:** www.pcsx2.net



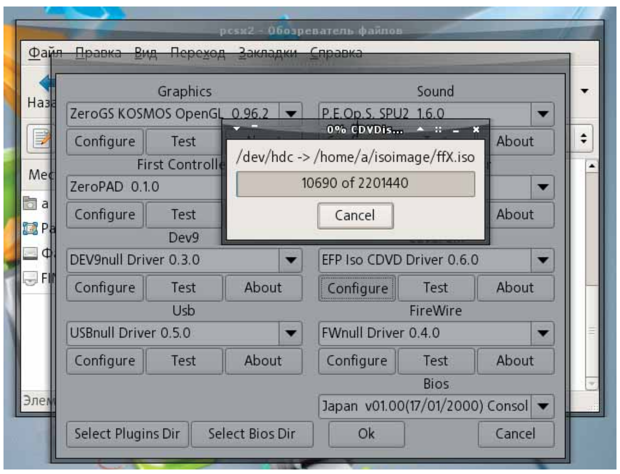
Если вы думали, что под эмуляторами можно играть только в ретро-игры, то, мои дорогие, вы жестоко ошибались. Хотите попробовать игры с ***SONY Playstation 2*** на вашем Linux? Тогда – за дело! На момент написания статьи для нас вышла специальная новейшая версия *PCSX2 0.93* с обновленными модулями дополнений (plugins). Пользователи Windows могут только молча завидовать, так как для них пока единственный вариант – довольствоваться старой версией. 0.93 – эксклюзив, выпущенный специально для Linux.

PCSX2 – это эмулятор, позволяющий реально играть в Playstation 2-игры на PC уже сегодня. Это – событие. Обратите внимание, с какой помпой в этом году был выпущен для PC-совместимых компьютеров порт *Resident Evil 4* – и вот игра работает на *PCSX2*! Как и *Final Fantasy X*! И *Tekken 4*!

Для нормальной полноскоростной эмуляции компьютера ***Amiga*** с 7-МГц процессором требовался Pentium 150 МГц при использовании эмулятора *xfellow* или 800 МГц с UAE, поэтому, надеюсь, вы понимаете, какой компьютер нужен для эмуляции 300+ МГц монстра. Пугаться особо не стоит, но очень рекомендуются процессоры с двумя ядрами (например, Pentium D 925 или Core 2 Duo).

Также для работы *PCSX2* (а точнее, видеомодуля ZeroGS KOSMOS OpenGL 0.96.2) вам потребуется специфическая библиотека libCG.so. Взять готовый RPM-пакет с ней можно на сайте NVIDIA.

Эмулятор использует для своей работы систему дополнительных модулей, отвечающих за поддержку различных устройств – вероятно, это лучший способ совершенствовать эмуляцию консоли. Отсюда вытекает, что если вы не удовлетворены работой какого-либо устройства, то попробуйте другой модуль.



Перед началом игры лучше снять образ с игрового диска. Для этого отлично подойдет модуль EFP Iso CDDVD Driver 0.60. Выбрав его, укажите в настройках, что хотите снять образ, и наберитесь терпения: это достаточно долгий процесс. Альтернативный вариант: воспользуйтесь стандартным *K3b*. Кроме того, обратите внимание на имя вашего DVD-устройства (внимание: игры с PS2 поставляются на DVD-дисках) – не исключено, что оно отличается от указанного в настройках. Обычно это **/dev/hdc**, но его можно точно определить, профильтровав вывод команды *dmesg* (dmesg | grep hd).

Ваша видеокарта должна поддерживать 3D-ускорение (проверьте его наличие: glxinfo | grep direct).

И, наконец, любителям *Beryl* (к коим отношусь и я) рекомендую на время работы эмулятора отключить трехмерный рабочий стол во избежание падения производительности.

После всего вышесказанного могу только добавить, что проект *PCSX2* находится в разработке уже 4 года, и с каждым выпуском количество играбельных шедевров увеличивается. *PCSX2* выпускается по лицензии GPL, то есть все факторы для весеннего оптимизма присутствуют.

Ну что же, бежим за охапкой дисков от PS2 и устанавливаем в привод! Перед покупкой не мешает ознакомиться со списком совместимости игр на сайте **PCSX2.net**. А *Final Fantasy X* или *Devil May Cry 3* хватайте, не раздумывая. Если какой-то файл пока не работает – не расстраивайтесь, а наберитесь терпения. Думаю, через год накопится уже внушительная подборка игр, нормально работающих в *PCSX2*.

Аркадные автоматы

Эмулятор: MAME **Сайт:** www.mame.net

Если вы соскучились по хорошей стрелялке, где от вас требуется только давить на курок, а шедевры типа *Hunt for Red October* или *Tyrian* давно пройдены и пылятся в глубине шкафа – обратите внимание на игры с аркадных автоматов.

Эти совершенно потрясающие устройства с невероятной мощью были нацелены на брутальное удержание игрока возле себя путем овладения его разумом через игры. Впрочем, мы это уже видели в фильме «Последний звездный боец», и если после прохождения всех уровней нагрянут инопланетяне – я вас предупреждал.

И снова пользователям свободных ОС здорово повезло. В нашем арсенале есть совершенно фантастический эмулятор гигантского количества игровых автоматов – *MAME*. Его разработкой занимаются, как водится, гениальные программисты. Релизы выходят один за другим: добавляются новые игры, улучшается эмуляция и скорость работы трехмерных игр. Можно даже сказать, что *MAME* – самый потрясающий эмуляционный проект в истории.

К сожалению, графического интерфейса *MAME* не предоставляет, но можно воспользоваться одной из оболочек. Кстати об оболочках: свою совместимость с *XMAME* заявляют многие, но разработка последнего практически заморожена. Активно развивается порт *SDLMAME*, его-то и имеет смысл установить к себе в систему.

Я собирал версию 0.112 из исходных текстов. Получившийся двоичный файл файл назывался **mamepm**. Чтобы запустить игру, достаточно набрать:

```
mamepm /путь к игре/игра.
```

Если в системе установлен *Beryl*, то разумнее задействовать мощь 3D-видеокарты для отображения игр и увеличить размер окна (менять размер окна можно прямо в запущенной игре).

Строка ниже запускает игру *Armed Formation* – ураганную космическую стрелялку:

```
./mamepm /путь к игре/armedf.zip -video opengl -window -maximize
```

Для игры в полный экран:

```
./mamepm /путь к игре/armedf.zip -video opengl
```

Количество поддерживаемых MAME игр приближается к пяти тысячам, и это реальные шедевры игростроения. Аркадные версии *Contra*, *Final Fight*, *Mortal Kombat* просто обязаны быть пройдены.

Думаю, я помог вам найти способ скоротать ближайшие выходные.

MS DOS / IBM PC

Эмулятор: DOSBox **Сайт:** dosbox.sourceforge.net

Ах, эти прежние деньки. Медленные компьютеры. *Comanche Overkill*, *Flashback*, *Jill of the Jungle*... *DOSBox* предлагает вернуть их, благодаря отличной эмуляции старых ***IBM PC***.

Под ним у меня запустилось больше игр, чем на настоящем 386SX 40МГц, но вам понадобится мощный компьютер. Если на столе антиквариат в духе Pentium III–500, то можно рассчитывать на приемлемую скорость работы программ, написанных для 286-х машин (*Prehistorik 2*, *Flashback*: тоже неплохо). Реально, желательно иметь 2000 МГц.

После запуска *DOSBox* на экране появится знакомый «черный экран» с приглашением **z:\>** внизу.

Это виртуальный диск, созданный самим «досбоксом». Его содержимое можно просмотреть, дав команду **dir** (не мучайте **ls**). Нам же интересно увидеть свои данные из домашней директории (например, из папки **games**). Для этого выполните команду:

```
mount c /home/пользователь/games
```

Появится виртуальный диск **C:**, включающий содержимое **games**. Перейдем на него:

```
c: <Enter>
```

Содержимое, как мы уже знаем, можно просмотреть командой **dir**. Я, к примеру, увидел у себя каталог **Gods** с игрой *Gods* и зашел в него командой *cd Gods* (автодополнение по TAB работает!), а затем запустил исполняемый файл, напечатав: **gods.exe**. Просто?

На самом деле проще было зайти в папку в вашем любимом файловом менеджере GNOME/KDE и щелкнуть на исполняемом файле, заранее ассоциировав его с *DOSBox*, но... Что если это CD-версия игры?

Если для игры необходим CD-ROM, дайте команду (при условии, что физический привод имеет точку монтирования **/media/cdrom**):

```
mount d /media/cdrom -t cdrom
```

Не сработает – попробуйте такой вариант:

```
mount d /media/cdrom -t cdrom -usecd 0
```

Но с *DOSBox* можно использовать и обычные ISO-образы. Не знаю, насколько это правильный путь, но всегда их снимал командой:

```
cat /dev/hdc >/home/user/disk.iso
```

и проблем не имел.

Монтирование ISO в *DOSBox* также не вызывает труда:

```
imgmount d /home/user/myiso.iso -t iso
```

В эмуляторе используются горячие клавиши:

Alt+Enter – переход в полноэкранный режим и обратно;



CTRL+F4 – смена подключенных ISO-образов;

CTRL+F5 – сделать снимок экрана;

CTRL+F6 – старт/стоп записи звука в файл WAV;

CTRL+F7 – уменьшить количество пропускаемых кадров;

CTRL+F8 – увеличить количество пропускаемых кадров;

CTRL+F9 – выход из эмулятора;

CTRL+F10 – включение мыши, если она автоматически не подключилась;

CTRL+F11 – замедлить эмуляцию;

CTRL+F12 – ускорить эмуляцию.

Для любителей покопаться в настройках есть файл **dosbox.conf**. В нем можно изменить многие параметры – в том числе объем памяти и качество звука. В принципе, стоит также включить использование OpenGL для борьбы с пикселизацией и получения гладкой картинки.

Надеюсь, после прочтения этого небольшого руководства вы сможете играть в любимые DOS'овские игры под Linux. Есть очень много отличных игр, те же *Jill of The Jungle* или *Jazz Jack Rabbit*, аналогов которым нет на других системах.

Сияй, безумный бриллиант

Завершая этот краткий экскурс, я хотел бы сказать, что коснулся лишь верхушки гигантского айсберга под названием «Эмуляция». Заметьте, мы пока даже не произнесли имена таких мегасистем, как ***SONY Playstation***, ***Nintendo 64***, ***NeoGeo***, ***ATARI Jaguar 64***, ***ATARI ST***, ***X68000***, ***PC98*** и даже существовавшей 32-битной приставки виртуальной реальности ***Virtual Boy***, ощутить объемность игр с которой можно и сегодня. Но об этом и многом другом – в следующий раз. А пока, надеюсь, вы прониклись идеей, что эмуляция – это еще одна грань, заставляющая безумно сиять бриллиант GNU/Linux.

Полезные ссылки:

www.zophar.net старейший эмуляционный сайт.

linuxemu.retrofaction.com эмуляционный Linux-сайт.

www.mameworld.net MAME-ресурс.

www.pbernert.com/ – сайт с модулями для PCSX2.

developer.nvidia.com/object/cg-toolkit-15#downloads библиотека Cg для ZSNES.



ETEGRO

LINUX FORMAT

Учебники



Наши эксперты помогут вам с любым приложением Linux



ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

Третья стадия?

Он похлопал Бога Солнца по плечу.
— Эй, солнышко!
Когда бог обернулся, Ом сломал рог изобилия об его голову.

Терри Пратчетт. Мелкие боги.

В мае-месяце случилось невероятное. Дело даже не в том, что ведомство Германа Грефа в лице его зама Андрея Шаронова раскритиковало безумный план закупки программного обеспечения для школ и рекомендовало обратить внимание на открытые программы вообще и на GNU/Linux в частности, а Министерство финансов прислушалось. Работа у Минэкономразвития такая – считать деньги, а Министерства финансов – их не давать. Удивительно, что даже и возразить этому никто ничего внятно не смог и, надеюсь, не сможет. Дело отнюдь не в отсутствии лоббистов – этого добра было хоть отбавляй. Но уж больно беспечно эти лоббисты себя вели – по-хозяйски. Само существование свободного программного обеспечения показало, что пресловутой монополии-то и нет. Конечно, везде стоит, как правило, сами знаете что, но никого не могут обязать и дальше сидеть на этой «игле». Альтернатива есть!

Однако само позиционирование как альтернативы – это, к сожалению, недостаточное условие для развития свободного сообщества. Нужны адепты в хорошем смысле этого слова. Чтобы появились тысячи, следует для начала «напрячься» для единиц. То есть – опять в школу. Оглянитесь вокруг и, если у вас есть родственники в образовательных учреждениях, помогите им на этом пути. Кстати, родственники среди учителей не обязательны – ученики тоже годятся. И если первых может и не быть, то вторых вполне можно завести.

E.M.Baldin@inp.nsk.su

В этом выпуске...























Совет месяца: Настольные жесты

Почти все **Советы месяца** так или иначе связаны с командной строкой. Все потому, что это универсальный инструмент, включенный во все дистрибутивы Linux, одинаково полезный как для профи, так и для новичков. Это не значит, что мы в LXF проводим свои дни в Bash – как и многие пользователи, мы предпочтем хороший GUI вводу пары-тройки команд. Просто **Советы** для графических инструментов писать сложнее – существует множество способов решения одной и той же задачи, и каждый предпочитает что-то свое. Но если и есть какая-то одна функция какой-то одной среды, которую незаслуженно забывают, то это панель Действия ввода в Центре Управления KDE. Она не бросается в глаза и вы, вероятно, никогда на ней не останавливались. Однако, десять минут, потраченные на знакомство с предлагаемыми опциями, могут изменить ваш рабочий стол.

Первое, что можно сделать с помощью Действий ввода – это назначить соответствие между жестами мышью и клавиатурными комбинациями. Представьте: легкий взмах курсором – и вы перемещаетесь между просмотренными web-страницами. А поскольку *Konqueror* – это не только браузер, но и файловый менеджер, вы можете применить данный подход и к локальной файловой системе.

Чтобы воплотить вышесказанное в жизнь, откройте панель Действия ввода – она находится в категории Региональные настройки Центра управления KDE. Выберите из списка действий Росчерки Konqueror и убедитесь, что флажок Выключено не установлен. Затем укажите требуемое действие (Вперед/Назад – хороший кандидат для первой попытки) и перейдите на вкладку Росчерки. Здесь необходимо изобразить мышью фигуру, соответствующую действию, при этом правую кнопку нужно держать зажатой. По нашему мнению прочерк слева направо хорошо подходит для функции Вперед, а справа налево – для Назад. Повторите каждый жест три раза, и постарайтесь, чтобы вышло похоже. Наконец, нажмите Применить и наслаждайтесь!

Система: Файлы

Что вам нужно? Файлы. Когда они вам нужны? Сейчас! **Энди Ченнел** покажет вам, как добраться до них из любого места...



Вы, поди, думаете: да ладно, не расскажет же он нам, как бесплатно загрузить полные версии *Spiritualized* или *Шрэк Третий*? И вы правы. Фраза «совместное использование файлов» имеет столь же безупречную репутацию, как Тони Сопрано, но мы говорим не о совместном использовании файлов в духе Kazaa или Gnutella, а о том, что называется «личное удаленное использование файлов». То есть получение доступа к вашим собственным файлам из любой точки мира.

Урок состоит из двух частей. В первой рассматривается то, что называется File Transfer Protocol (FTP), как со стороны сервера, так и клиента, а во второй, более короткой, мы рассмотрим кое-какие методы хранения файлов и документов в сети, чтобы не настраивать каждый раз ПК, а то и держать его дома постоянно включенным, для возможности доступа к вашей информации в дороге. Оба этих подхода к личному удаленному использованию файлов полезны. Один хорош, когда у вас много файлов, которые должны быть постоянно под рукой, а другой выгоден, когда ваши нужды не требуют много места – хотя, используя данные методы, вы можете хранить в сети до 1 ГБ – но вы предпочитаете упрощать себе жизнь. Оба подхода имеют свои плюсы и минусы, если вы хотите использовать свои файлы совместно с друзьями и коллегами (коль скоро у вас есть права на раздачу этих файлов), и я упомяну их по ходу рассказа.



Наш эксперт

Энди Ченнел
Энди делает свои первые шаги в Linux уже шесть лет, а технологиями интересуется еще со времен Dragon 32

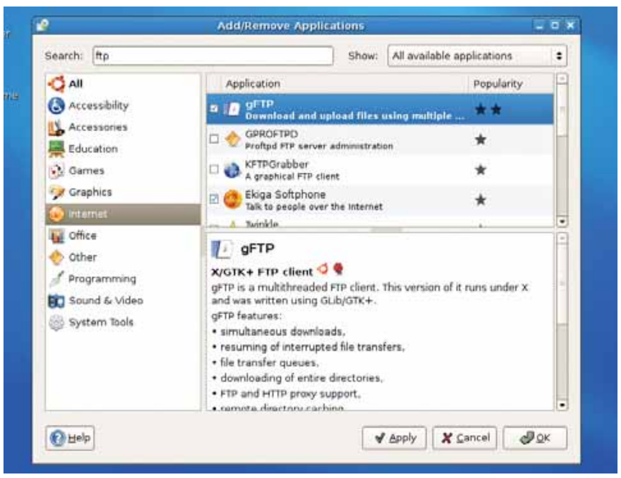
Часть 1: Совместное использование по FTP

Как и многие системы передачи информации, FTP основан на технологии клиент-сервер. Это означает, что в любой транзакции участвуют два элемента: сервер (приложение, запущенное на удаленное машине) и клиент (приложение, запущенное на локальном компьютере). Мы соединяем последний с первым, чтобы посылать и принимать данные. Поскольку более вероятно, что мы будет больше времени тратить на доступ к файлам, чем на предоставление доступа, то сначала рассмотрим клиент. При помощи этого ПО вы можете подключаться к удаленному серверу и работать с файлами так (в соответствии с правами доступа к серверу), как будто они находятся у вас на локальной машине.

Для Linux имеется множество клиентов, как графических, так и для командной строки. Мы, неизбежно, заинтересованы в графических клиентах, наиболее яркие примеры которых – *GFTP*, *IglooFTP*, *KFTPGrabber* и *Konqueror*. На нашем уроке я использую Ubuntu, но эти приложения доступны для многих популярных дистрибутивов. Вы вполне можете установить подходящий клиент при помощи вашего обычного менеджера пакетов. Именно так я выбрал GFTP (просто потому, что это наиболее популярный клиент для Ubuntu), но процесс не зависит от того, какой пакет вы решили установить.

В Ubuntu выполните Приложения > Добавить/Удалить и просмотрите страницу Интернет в поисках соответствующих программ. Выберите их, щелкнув на флажке, рядом с ними, щелкните на кнопках Применить и OK и подождите, пока приложение загрузится и установится. По завершении вы сможете найти приложение в меню Приложения > Интернет. Запустите его.

Большинство FTP-клиентов в главном окне имеют две панели, плюс стандартную панель инструментов. В *GFTP* левая панель отображает локальную файловую систему, а правая – удаленную файловую систе-



› **В вашем дистрибутиве должно быть много FTP-клиентов..**

» **Месяц назад** Мы осветили темные углы и секреты файловой системы Linux.

в общем доступе

На DVD » GFTP 2.0.18

Исследуем интерфейс GFTP

1 Параметры соединения

Здесь вы настраиваете сервер, к которому подключаетесь. В данном примере я соединяюсь с FTP-сайтом *Gimp*, чтобы установить последнюю версию. Здесь размещены (слева направо): кнопка Соединение/Отключение, адрес сервера, номер порта (обычно здесь стоит 21), имя пользователя и пароль, а также выпадающий список для определения протокола соединения. Наконец, у правого края находится кнопка для полной остановки любой активности.

2 Локальная файловая система

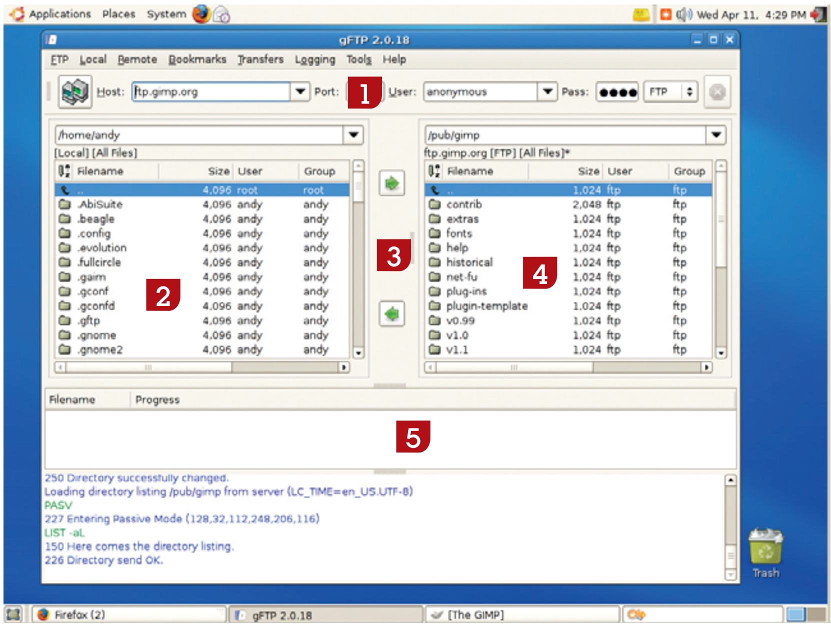
Это работает так же, как и файл-менеджер вроде *Nautilus* или *Konqueror*. Вы можете, если хотите, копировать и вставлять файлы внутри этой системы обычным способом.

3 Кнопки Разместить [Push] и Скачать [Pull]

Если протащить и бросить слишком трудно, то выделите файл, который хотите передать в локальную или удаленную файловую систему, а затем нажмите соответствующую кнопку для его размещения на сервере или загрузки в вашу файловую систему.

4 Удаленная файловая система

Это файловая система сервера, к которому вы подключились. Хотя она выглядит точно так же, как и локальная, ваши возможности изменять что-либо ограничены правами, которые определяет администратор. Так, например, для FTP-сервера *Gimp* мы свободно можем скачивать файлы с сервера, но не можем что-либо разместить на нем и не можем изменять саму структуру файловой системы.



5 Панель активности

В основном, это журнал всех действий, выполняемых клиентом. Это то место, куда следует посмотреть в первую очередь при возникновении проблем, и он также полезен, если вы хотите точно знать, что вы, собственно, делаете.

му (если вы соединены с удаленным сервером). В этих панелях можно перемещаться так же, как и в обычном менеджере файлов. Например, дважды щелкните на каталоге, чтобы просмотреть его содержимое; чтобы вернуться назад к дереву, дважды щелкните на двух точках (..) в верхней части окна. Хотя эти окна отображают разные вещи, между ними можно перетаскивать файлы: так, в верхней врезке (на которой показан *GFTP*, соединенный с FTP-сервером *Gimp*), я могу перенести последнюю версию *Gimp* (или любой другой файл) напрямую в любой каталог моей локальной файловой системы точно так же, как с одного диска на другой. На самом деле, именно это я и делаю, только вместо путешествия по шлейфам в моем ПК, файлы перемещаются в сети Интернет.

FTP-клиент во многом схож с web-браузером. Вы вводите адреса и просматриваете документы, только они не отформатированы для отображения, как это сделано на web-страницах. *GFTP* даже имеет функцию закладок, которые вы можете использовать, чтобы вернуться к часто посещаемым сайтам (например, на ваше персональное место для разделяемых файлов), не вводя каждый раз все детали.

Размещение файлов

Установив и настроив клиент, разберем другую сторону, то есть сервер. Поскольку Linux заслужил свою репутацию на обслуживании файлов, то вы можете ожидать большого числа доступных FTP-серверов, и будете правы. Однако вместо выбора одного из стандартных пакетов типа *PureFTP* или *GProFTPd* я собираюсь обратиться к небольшому приложению под названием *CrossFTP* (www.crossftp.com). В отличие от стандартных FTP-серверов в Linux, это приложение изначально разрабатывалось как графическое, поэтому имеет очень приятную оболочку, используемую для настройки и управления. К тому же *CrossFTP* является полностью кроссплатформенным, поскольку разрабатывается на Java и выпускается по открытой лицензии Apache.

CrossFTP устанавливается прямо с web-сайта проекта, и, если ему позволить, размещает ярлык на рабочем столе. Дважды щелкните на нем для запуска графического интерфейса. После этого ныряйте в приложение и жмите кнопку Запустить сервер [Start Server] – и если у вас Windows, он, вероятно, заработает. Однако Linux больше заботится о безопасности и резервирует порты с малыми номерами (то есть те, через которые к вам могут подключиться другие компьютеры) для пользователя root [это не совсем так: подключение возможно к любому порту, но только приложениям, работающим от имени root, дозволено использовать порты с номерами меньше 1024 – а именно в этот диапазон попадают стандартные порты большинства популярных служб, в частности, FTP, – прим. ред.]. Как я уже говорил, FTP обычно соединяется по порту 21, и вы должны быть суперпользователем, чтобы запустить сервер с этими настройками. Дабы не заниматься ерундой, намного проще выбрать порт соединения с большим номером, например, 8021. Чтобы сделать это, нажмите кнопку Настройка [Configure] в главном окне и введите '8021' в строке Порт [Port]. Для закрытия окна настроек нажмите OK, а затем попытайтесь вновь нажать Запустить сервер [Start Server]. Получилось! »

» Следующий этап – добавление одного или нескольких пользователей и назначение им паролей и домашних каталогов. FTP-серверы обычно запирают пользователей «в песочнице», чтобы они не могли получить доступ к внешним каталогам, в частности, настройкам сервера. К тому же, добавляя имена пользователей и пароли, мы повышаем шансы защитить наши файлы от чересчур любопытных.

В левой части окна *CrossFTP* выберите вкладку Пользователи [Users]. Создадим одного пользователя на базе основного пользователя этой локальной машины, а внутри его каталога **/home** – каталог специально для разделяемых файлов. Для начала запустите ваш обычный менеджер файлов, перейдите в **/home/username** и создайте каталог с именем вроде **/share**. Теперь вернитесь в *CrossFTP* и введите имя пользователя и пароль, да посильнее, поскольку каталог может быть доступен из Интернета. Прямо под строкой Корневой каталог [Root Directory] находится кнопка Обзор [Browse]. Щелкните на ней, а затем через окно выбора файлов перейдите к желаемому каталогу. Нажмите кнопку Разрешить [Enabled], чтобы сделать его активным, затем, если вы собираетесь изменять файлы в каталоге удаленно, установите опцию Права на запись [Write Permission]. Можно, конечно, оставить эту опцию не установленной, если вы собираетесь при удаленном доступе только читать файлы или создаете открытый каталог для работы коллег или широких масс. Введя все что полагается, нажмите кнопку Сохранить [Save], и на этом все. Вы готовы к подключению.

Скорая помощь

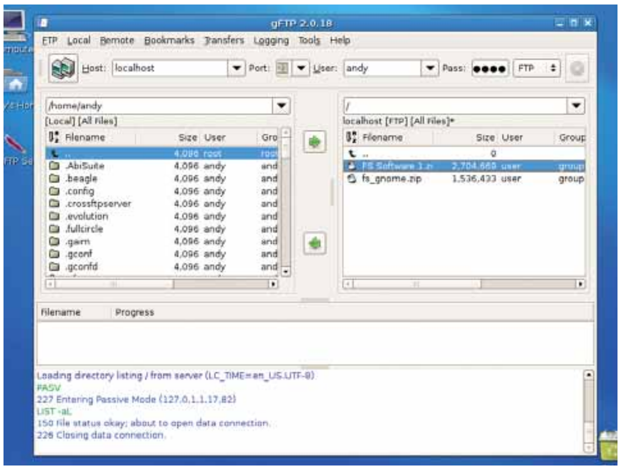
Если вы храните документы в сети или открыли доступ к ним на вашем ПК из сети, чрезвычайно важно иметь сильный пароль в виде комбинации букв и цифр, желательно не составляющей словарное слово. Можете также предусмотреть и шифрование.

Собрать все вместе

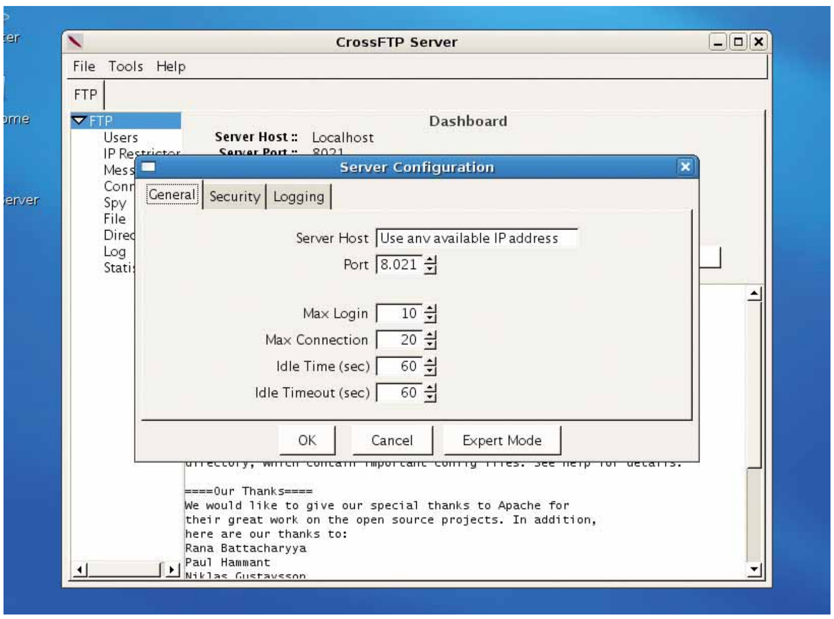
Для тестирования, создадим локальное соединение. Во-первых, возьмите пару файлов и поместите их в ваш каталог **/share**. Теперь запустите установленный и настроенный ранее FTP-клиент (*GFTP*). Добавьте следующие параметры: Host: localhost, Port: 8021, User: имя пользователя в соответствии с произведенными настройками, Password: пароль. Теперь нажмите кнопку Соединение [Connect], и – с Богом... Слева вы должны увидеть всю свою файловую систему, а справа – содержимое каталога **/share**.

Хотя мы и подключились локально, процесс удаленного подключения практически таков же, за исключением нескольких принципиальных отличий и чуть большей сложности. Эти сложности обусловлены двумя вещами: безопасностью и принципом работы Интернета. Во-первых, следует настроить ваш брандмауэр, разрешив подключение к порту 8021. Как это сделать, зависит от того, программный ли у вас брандмауэр или он является частью маршрутизатора. В *Firestarter*, популярном брандмауэре для Linux, вам необходимо произвести изменения внутренней политики, путем Добавления (Add) Новой службы (New service) и указания, что это FTP, номер порта 8021 и доступ разрешен всем (Anyone), то есть разрешить подключение по этому порту, минуя брандмауэр. Схожую процедуру необходимо проделать и с аппаратным брандмауэром, но для этого следует обратиться к прилагаемой документации. Например, в моем маршрутизаторе Linksys нужно указать номер порта, намеченного к открытию, а затем определить компьютер в сети (по IP-адресу), к которому следует перенаправлять эти подключения.

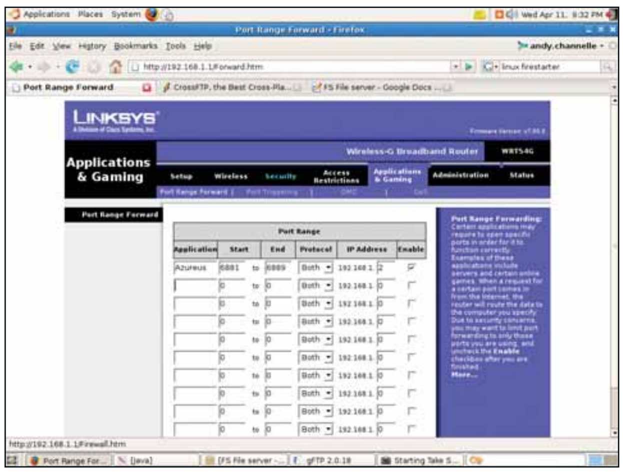
Другая область, на подробное обсуждение которой в данной статье нет места – это динамический DNS. Большинство интернет-провайдеров предоставляют IP-адрес – это уникальный адрес для вашего компьютера – на основе общего пользования, так что адрес 82.167.55.03 сегодня у вас, а завтра перейдет к другим клиентам. Это изрядно затрудняет подключение к вашему частному серверу. Выходов здесь два. Первый – запросить у провайдера статический IP-адрес; но провайдер может за это дополнительно подоить ваш кошелек. Второй, более дешевый способ – использовать динамический DNS, подписавшись на службу динамической пересылки IP-адреса, типа No IP (**www.noip.com**). Вам предоставят приложение, которое будет запускаться при каждой загрузке, записывать текущий IP-адрес и отсылать его серверу имен. Вторая часть этой службы предоставляет вам доменное имя (что-то типа **bob-smith.hopto.org**), связанное с информацией, присылаемой на сервер первой программной частью. Теперь вы сможете использовать ваш FTP-клиент для входа на **ftp.bob-smith.hopto.org**, а служба автоматически перенаправит вас к верному IP-адресу, и вы доберетесь до своих файлов.



› *GFTP* (клиент) успешно подсоединился к *CrossFTP* (сервер), и общая папка стала доступной.



› У *CrossFTP* простой интерфейс, и он работает в любой ОС, способной выполнять Java.



› Неважно, аппаратный у вас брандмауэр или программный – вы не сможете подключиться, пока не настроите все правильно.

Часть 2: Использование сетевых сервисов

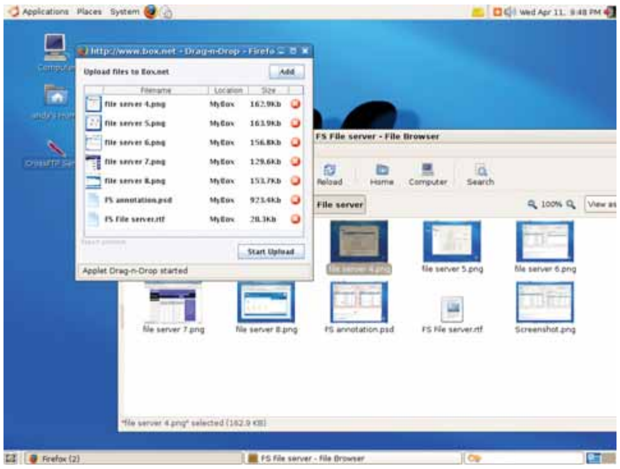
Если вы часто обращаетесь к файлам с вашего домашнего ПК, то FTP – хорошее решение; но если вам изредка необходим доступ к нескольким документам, то они могут быть скормлены какому-нибудь сетевому хранилищу, и тогда будут доступны с любого ПК. Сетевые хранилища существуют уже десятилетия, но лишь недавно хранение и трафик настолько подешевели, что это стало реальным вариантом. Более того, Web 2.0 как таковой означает, что вы сможете получить доступ к вашим документам и даже редактировать их, не покидая уютного *Firefox*.

Нечего и говорить, что такой тип сервиса предполагает наличие широкополосного доступа в Интернет – передача файлов при помощи старого доброго dial-up, похоже, будет весьма утомительной. Важно также не забывать о безопасности, то есть обеспечить разумные имя пользователя и пароль и не размещать важную информацию вроде финансовых документов или компрометирующих изображений (вас и секретарши) в Сети.

Я выбрал для использования **www.box.net** по ряду причин: у него есть несколько интересных функций кроме обычного хранения; похоже, он переживет следующий крах дот-комов; и он бесплатен (о других опциях см. наше Сравнение на стр. 18). Первым делом посетите сайт и зарегистрируйтесь. Сервис предлагает 1-ГБ хранилище бесплатно, с ограничением на размер индивидуальных файлов в 10 МБ и общий месячный трафик в 10 ГБ. Вы платите за расширение хранилища и снятие ограничений на трафик, за возможность открывать дочерние учетные записи и отмену ограничения скорости; однако бесплатный сервис вполне сгодится большинству пользователей – включая нас.

После регистрации и входа, вы будете перенаправлены к «ящику» ['box'] с именем MyFiles, в котором вам и следует хранить свои файлы. Очевидно. Внутри этого пространства вы можете выполнять обычные для файл-менеджера операции: создание и удаление каталогов и файлов, их переименование, перемещение и поиск.

Давайте добавим несколько файлов. Имеется несколько способов сделать это. Первый и простейший – щелкнуть на ссылке Upload New Files [Выгрузить новые файлы], что предоставит вам на выбор два варианта. Первый – щелкнуть на кнопке Add [Добавить], это приведет к выводу стандартного окна выбора файла. Вы можете пометить несколько файлов подряд (то есть стоящих в списке один под другим), щелкнув на первом файле, или выбрать разрозненные файлы, используя клавишу Ctrl и мышь, и затем вновь щелкнуть на кнопке Выбрать [Select] для загрузки их в ваш «ящик». Более интересная опция – выбор Drag-n-Drop в диалоговом окне. При этом запускается небольшой Java-апплет, который вы можете сохранить в активном состоянии на вашем рабочем столе, что позволит вам просто перекидывать файлы прямо в его окно. Нажмите кнопку Добавить [Add], и они будут загружены автоматически. Оба этих способа загружают файлы в каталог, для которого они были применены.



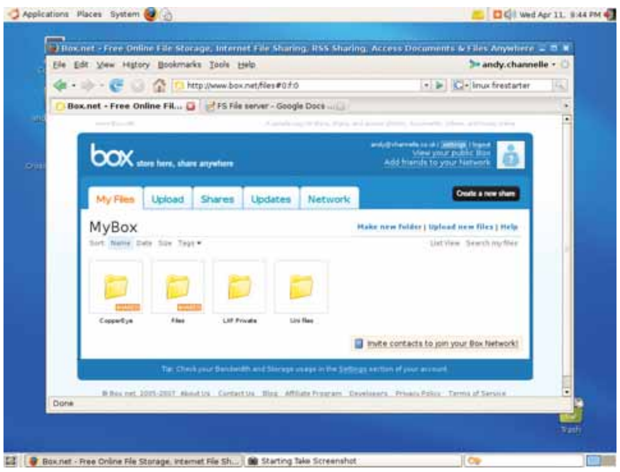
› Для быстрой загрузки ваших файлов весьма полезно держать Java-апплеты в активном состоянии.

Есть еще и третий способ, который хорош, если вы в дороге: отошлите файл по почте на адрес **upload@box.net** из-под вашей учетной записи, и он будет добавлен в папку Upload From Email [Выгружено по e-mail]. Позднее, при следующем входе, вы сможете переместить его в соответствующее место.

Сервис **Box.net** оснащен несколькими весьма интересными опциями совместного использования файлов зарегистрированными пользователями и случайными посетителями, и начал вводить полезные функции редактирования, которые позволяют, например, открыть текстовый документ в текстовом online-процессоре *Zoho Writer* и затем сохранить его в вашем ящике.

Какой же метод лучше: FTP или сеть? Полностью настроенный FTP-сервер не только обеспечивает больший контроль над вашими данными с точки зрения безопасности, но также заставляет лучше понять, как взаимодействуют ваша ОС и окружающий мир. Сетевые хранилища обеспечивают быстрый результат и, возможно, за ними будущее, когда вся «текущая работа» переберется в сетевые репозитории.

Независимо от вашего конечного выбора – а может, вы выберете комбинированный вариант – вы получите приятное ощущение того, что ваши важные документы находятся на расстоянии одного щелчка. LXF



› **Ваш ящик [box] имеет несколько функций стандартного файл-менеджера – у всех бывают скрытые таланты.**

Хочу предупредить...

Сетевые хранилища – это здорово, но всегда существует шанс, что компания, поддерживающая сервис, прогорит или откажет в бесплатных услугах. Вот несколько правил, которые стоит соблюдать, чтобы не остаться у разбитого корыта.

1. Выберите авторитетный сервис. Например, Google или Yahoo! весьма стабильны. Если вы должны работать с новичком, посетите форум компании и посмотрите отзывы других.
2. Рассматривайте это как кратковременное хранилище. Размещайте документы, которые могут понадобиться вам для конкретного задания, и регулярно выполняйте оффлайн-резервирование.
3. Постарайтесь объединить оба описанных метода. Используйте защищенные разделы платного web-хостинга для хранения данных в сети и работайте с ними при помощи FTP-клиента.
4. Подумайте о платных службах. Иногда вы действительно получаете что-то очень нужное за свои деньги.

» **Через месяц** Научимся создавать классные меню для ваших DVD.

Calc: Макрос для

Перелопатили все биржевые котировки, только чтоб убедиться: BMW вам уже не купить? **Марк Бэйн** упростит вам жизнь...





Наш эксперт

Марк Бэйн
Свою первую программу Марк написал еще для ZX81. Он много лет работает с Unix и Oracle, а также читает лекции и пишет книги.

Надеюсь, ваши акции поживают лучше, чем мои – если нет, вам остается только ронять слезы в пиво. Но не падайте духом! Здесь в Linux Format мы, может, и не улучшим ваши навыки по прогнозированию котировок, зато вы с ходу будете видеть, насколько плохи ваши дела.

Поможет, конечно же, *OpenOffice.org*. На нашем уроке мы сперва рассмотрим, как загрузить финансовую статистику с сайта прямо в таблицу Calc; затем – как по этим данным построить график; а в завершение научимся все это делать автоматически. Фактически, вы сумеете это делать, даже не открывая *OOo* (кроме как для просмотра результатов). Вообще-то, говоря, что вам не придется открывать *OpenOffice.org*, я имею в виду, что нужно будет написать пару макросов – но затем вы и вправду сможете обрабатывать данные без открытия *OOo*.

В любом случае, перво-наперво найдем финансовую статистику. У вас могут быть свои источники – ну, а мы воспользуемся финансовым порталом Yahoo Finance. Если вам безразличны финансовые сводки, не спешите захлопнуть страницу: навыки, полученные здесь, применимы и к другим источникам данных. Однако для целей нашего урока, алчность – это хорошо!

Получаем данные

Вы найдете Yahoo Finance по адресу **http://finance.yahoo.com**, им довольно легко пользоваться – там есть окошко, содержащее слова Enter Symbol(s) [Введите Символ(ы)] и кнопка с названием Get Quotes [Получить котировки]. Все, что вам надо сделать, это ввести символ компании, в которой вы заинтересованы; Yahoo Finance предоставляет ссылку, которая даст вам этот символ. Например, символ Microsoft на Лондонской фондовой бирже – MSFL. Как только вы введете этот символ и нажмете Получить Котировки, вы увидите самые последние данные о фирме. А сейчас начнется самое интересное: скачаем их!

Взглянув на экран котировок, вы увидите ссылку Скачать Данные [Download Data]. По ней мы добудем URL самих данных: **http://download.finance.yahoo.com/d/quotes.csv?s=MSF.L&f=sl1d1t1c1ohgv&e=.csv**. А зачем нам URL? Затем, что теперь можно в любой момент набрать его в браузере. Времени тут экономится немного, но главное – начать.

На следующем шаге мы сэкономим гораздо больше времени. Если вы нажмете на Скачать Данные или напечатаете URL, *Calc* загрузит CVS-файл в мастер импорта. С его помощью вы можете указать разделители данных в файле *OpenOffice.org* (в нашем случае это, очевидно, будут запятые), а затем *OOo* поместит каждое поле в свой столбец. Пока ничего нового – вы, возможно, проделывали такое уже сотни раз. Но вместо ручного ввода лучше написать макрос.

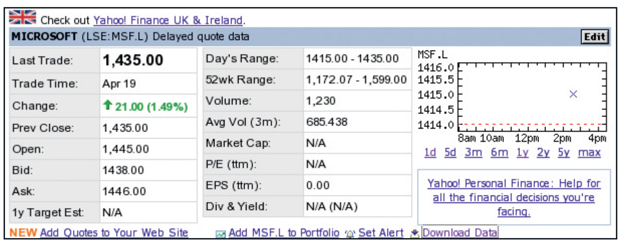
Первый макрос

Вы уже поняли, что можно получить котировки акций, написав символ, представляющий фирму, как часть URL в вашем браузере, а затем загрузить их как CSV-файл в *OOo Calc* через мастер импорта. Теперь, прежде чем писать макрос для импорта, проанализируем импортируемые данные. Заметьте, что поля, которые скачиваются, определяются параметром 'f' в URL. Поэкспериментировав с ними, вы поймете, как работает каждое из них, но сейчас нам нужны только три: 's', название фирмы, 'l1', последняя цена продажи, и 'd1', последняя дата сделки. Таким образом, наш URL превращается в **http://download.finance.yahoo.com/d/quotes.csv?s=MSF.L&f=sl1d1&e=.csv**, его мы и будем использовать в макросе.

OK, пора писать макрос. Откройте *OpenOffice.org* (неважно, какое приложение – сгодится *Calc* или *Writer*), затем нажмите Tool > Macros > Organize Macros > OpenOffice.org Basic. Здесь есть список языков на выбор (Basic, Python, BeanShell или JavaScript), но нам сейчас нужен Basic. Появится диалоговое окно, в нем надо нажать на кнопку Organizer. Откроется другой диалог, со вкладкой Module; выберите пункт Standard. Вы, небось, уже недоумеваете: да зачем вообще? – а дело-то простое. *Standard* – библиотека; в библиотеках хранятся модули; в модулях хранятся макросы. Дошло? Тогда жмите на New, чтобы создать свой собственный модуль в библиотеке *Standard*.

Нажав на New, вы увидите, что *OOo* предложит имя вашему модулю: 'Module1'. Не используйте его. Довольно нудно работать с модулями, названными Module1, Module2, Module3, и так до бесконечности. Лучше дать модулю значимое имя – например, свой я назвал 'lxf94' (угадайте, почему). В новом модуле нажмите Edit, и *OOo* откроет экран редактирования Basic – именно тут мы и создадим макрос.

Тут вы обнаружите, что *OpenOffice.org* уже создал для вас макрос



› Скачайте с Yahoo Finance в таблицу *OpenOffice.org* котировки акций в виде CSV-файла.

LINUX FORMAT На DVD
» Код примеров статьи

чтения данных

Используем crontab

Если вы новичок в *crontab*, то поля и звездочки могут вас озадачить. Однако порядок полей довольно прост:

1. **Минуты** (0–59)
2. **Часы** (0–23)
3. **Число** (1–31)
4. **Месяц** (1–12)
5. **День недели** (0–6, где 0 означает воскресенье)
6. **Запускаемая команда**

Звездочка означает «запускать в любом случае» – так, * в третьем поле означает «запускать каждый день месяца». Вы также можете использовать комбинации чисел в полях: 1,3,10–20 (снова в 3-м поле) будет означать «запускать по первым и третьим числам и каждый день с 10-го по 20-е».

с именем Main. Предупреждаю: не путайте его с Main, знакомым вам по другим языкам программирования. Здесь это просто пустой макрос, чтобы *OpenOffice.org* было с чем работать, когда вы нажмете кнопку запуска (на ней изображены угол страницы и стрелка, указывающая вниз). На самом деле *OOo* запустит первый же макрос, который найдет в модуле. Например, если поместить Main1 выше Main, то он и будет запущен.

Вооруженные этими знаниями, мы готовы писать макрос (поместите этот код после процедуры Main):

```
Function open_csv_file (url as String) as Object
Dim oProperty(0) as New com.sun.star.beans.PropertyValue
oProperty(0).Name = "FilterOptions"
oProperty(0).Value = "44"
open_csv_file = starDeskTop.loadComponentFromUrl(url, "_
blank", 0, oProperty())
End Sub
```

Созданная функция загружает любой CSV-файл без всякого мастера импорта, используя свойство FilterOptions при установке его значения в 44. Почему 44? Потому что 44 – это код ASCII для… верно, для запятой. А зачем здесь функция? Затем, что open_csv_file создает объект – сам документ – а нам потребуется к нему доступ из кода.

Если вы теперь попробуете запустить код, то ничего не произойдет – потому что Main пока ничего не содержит. Заставим его поработать:

```
Sub Main
Dim oDoc as Object
oDoc = open_csv_file _
("http://download.finance.yahoo.com/d/quotes.
csv?s=MSF.L&f=sd1l1&e=.csv")
End Sub
```

Теперь при нажатии на кнопку запуска произойдет следующее: после пары секунд ожидания (в зависимости от скорости соединения) появится файл **Calc**, содержащий последние цены на акции для Microsoft на Лондонской фондовой бирже. Уже неплохо, но вы явно думаете «Не проделывать же это для каждой из моих акций – хорошо бы отобразить сразу весь мой портфель.» Отлично, этим и займемся.

Макрос для групповухи

Мы уже видели, как легко автоматизировать загрузку данных по одной фирме. Чтобы сделать тоже самое для их группы, надо просто разжиться новым символом и добавить его в URL. Например, если нас интересует Novell, добавим символ NOVL; наш URL станет таким: http://download.finance.yahoo.com/d/quotes.csv?s=MSF.L&s=NOVL&f=sd1l1&e=.csv. Ради этой новой информации модифицируем наш код. Начнем с добавки новой процедуры – она упростит нам жизнь:

```
Sub download_stock_price(companySymbols)
Dim oDoc as Object
Dim cSymbols as String, oUrl as String
cSymbols = join(companySymbols, "&s=")
oUrl = "http://download.finance.yahoo.com/d/quotes.csv?s=" _
& cSymbols & "&f=sl1d1&e=.csv"
oDoc = open_csv_file(oUrl)
End Sub
```

Вы заметите, что новый макрос принимает companySymbols как параметр – это будет массив, содержащий список кодов фирм.

Процедура соединяет все символы в единую строку (с &s= между каждым символом), а затем создает правильный URL для заготовленной нами функции open_csv_file. После создания нового макроса изменим процедуру Main, чтобы она принимала массив фирм:

```
Sub Main
download_stock_price(array("MSF.L","NOVL"))
End Sub
```

Когда на этот раз вы нажмете кнопку запуска, то увидите файл *Calc*, содержащий детали всех фирм, указанных в списке – и, конечно, можете добавить столько символов фирм, сколько душе угодно.

Сохраняем файл

Следующий этап автоматизации – сохранение файла на жесткий диск. Чтобы это сделать, опять напишем макрос:

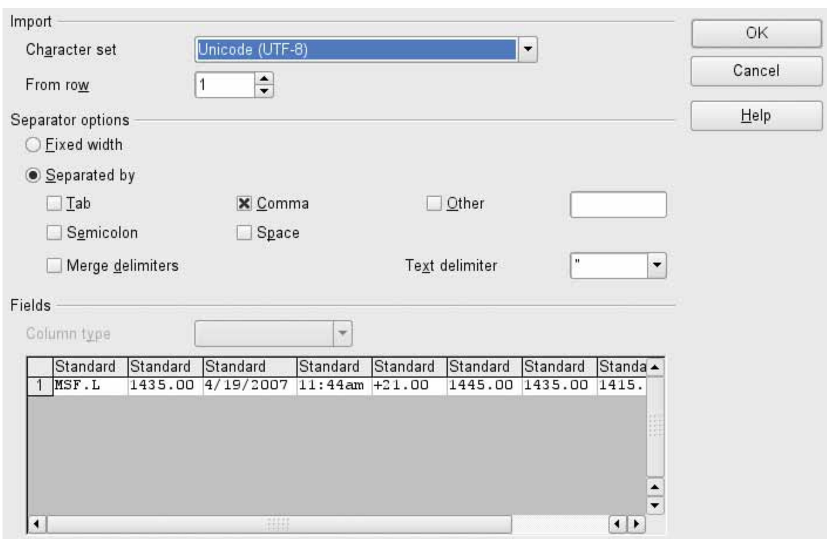
```
Sub save_file_as_ods(doc as Object, directory as String, prefix as
String)
Dim oUrl as String
oUrl = convertToUrl(directory & "/" & prefix & ".ods")
doc.storeAsUrl(oUrl,Array())
End Sub
```

Здесь следует обратить внимание на пару моментов: во-первых, на функцию convertToUrl. При открытии и закрытии файлов макросу требуется имя файла в определенном формате – и convertToUrl делает для вас это преобразование. Во-вторых, вы увидите, что наравне с ка- »

Скорая помощь

На этом уроке мы использовали текущие котировки акций, но ничто не мешает исследовать исторические данные для каждой фирмы.

› Без макроса вам придется проходить через мастер импорта *Calc*.



» талогом и префиксом файла (то есть именем файла без расширения '.ods') процедуре также передается doc. Это имя объекта, созданного с помощью функции open_csv_file.

Далее вам надо немного изменить функцию download_stock_price. Добавьте следующую строку в конец процедуры:

```
save_file_as_ods(oDoc, "/tmp", "test_lxf94")
```

На этот раз, запустив Main, вы увидите, что ваш документ переименован в **test_lxf94**, а если вы заглянете в каталог **/tmp**, то найдете файл **test_lxf94.ods**, который (если его открыть) покажет свежескачанные данные.

В командную строку!

В начале этой статьи я говорил, что все это можно делать, не открывая *OpenOffice.org* (кроме как для просмотра результата). Давайте выясним, как этого достичь. Узнав, как автоматически открывать и сохранять наш файл, вы сумеете это сделать и без отображения результатов. Начнем с добавления строки кода в конец макроса download_stock_price:

```
oDoc.close(true)
```

Она всего-навсего закрывает документ. Если вы запустите Main, то увидите, что документ откроется и сразу же закроется. Зачем мы начали с этого? Да просто чтобы удостовериться, что у нас не останется никаких сессий в фоновом режиме, когда мы приступим к изменениям, которые спрячут наш документ.

Вы уже знаете, как открыть CSV-документ, не используя мастер импорта, установкой соответствующих свойств, и, по-моему, нетрудно сообразить, что открытие документа в невидимом режиме происходит точно так же. Итак, назад к функции open_csv_file. Сначала переопределим свойства, намеченные для использования:

```
Dim oProperty(1) as New com.sun.star.beans.PropertyValue
```

Затем укажем детали для нового свойства:

```
oProperty(1).Name = "Hidden"
oProperty(1).Value = True
```

Проделав изменения в коде, нажмите на кнопку запуска... и опять ничего. Не пугайтесь. Проверьте:

```
bainm@aeneas:~> ls -l /tmp/test_lxf94.ods
-rw-r--r-- 1 bainm users 6455 2007-04-23 09:00 /tmp/test_lxf94.ods
```

Теперь запустите макрос и снова проверьте. Вы увидите, что файл действительно изменился:

```
bainm@aeneas:~> ls -l /tmp/test_lxf94.ods
```

Скорая помощь

Можно быстро добраться до Органайзера, нажав на Tools > Macros > Organize Dialogs, а затем перейдя на вкладку Module.

Необязательный ввод

На нашем уроке мы отключали Скрытый режим, но если вы собираетесь делать это довольно часто, то можете поменять код, чтобы сделать этот аргумент необязательным:

```
Function open_csv_file (url as String, Optional show_form as
Boolean) as Object
Dim oProperty(1) as New com.sun.star.beans.PropertyValue
if ismissing(show_form) then
show_form = False
end if
oProperty(0).Name = "FilterOptions"
oProperty(0).Value = "44"
oProperty(1).Name = "Hidden"
if (show_form) then
oProperty(1).Value = False
else
oProperty(1).Value = True
end if
open_csv_file = starDeskTop.loadComponentFromUrl(url, "_
blank", 0, oProperty())
End Function
```

Только помните, что по умолчанию документ останется невидимым, а если вы захотите его все же узреть, просто измените код в open_csv_file на

```
oDoc = open_csv_file(oUrl, True)
```

```
-rw-r--r-- 1 bainm users 6454 2007-04-23 09:01 /tmp/test_lxf94.ods
```

Это значит, что вы можете запускать процессы незримо, но (в настоящий момент) надо держать *OpenOffice.org* открытым, чтобы запускать макрос. А надо ли? Нет! Закройте все экземпляры *OOo* и наберите в командной строке:

```
scalc -headless "macro:///Standard.lxf94.Main"
```

Абсолютно ничего не произойдет – вы даже не увидите экрана *OpenOffice.org*. Однако если вы запустите *ls -l* и посмотрите на файл, то заметите перемены.

Для завершения процедуры автоматизации, создадим задачу *Cron*: тогда уже не придется беспокоиться о запуске макроса. Например, поставим задание *Cron* на запуск каждый день в 8:45, и в 9 утра к вашему приходу уже будет новый файл. Чтобы это сделать, откройте командную строку. Наберите *crontab -e*, а затем:

```
45 8 * * * scalc -headless "macro:///Standard.lxf94.Main"
```

Далее нажмите Control-D, чтобы сохранить файл *Cron*. И все – загрузка ваших котировок полностью автоматизирована.

Представляем графики

Каждый знает, что «лучше один раз увидеть»… это же касается и графиков. Не знаю, как насчет вас, но я понимаю графики лучше, чем голые цифры – а у нас пока что есть только цифры. Давайте преобразуем данные.

Вы ведь умеете создавать график в *OpenOffice.org*, правда? Просто выбираете ячейки, которые хотите использовать в качестве диапазона, нажимаете на Insert Chart и следуете инструкциям – нет ничего проще. Но разве это не утомительно? Я думаю, вы согласитесь, что это еще один кандидат на автоматизацию. Очевидно, не помешает создать процедуру для рисования графика, но прежде чем это сделать, немного изменим исходный код.

Для начала, мы не будем закрывать документ в процедуре download_stock_price:

```
REM oDoc.close(true)
```

Затем перестанем прятать документ в open_csv_file:

```
oProperty(1).Name = "Hidden"
oProperty(1).Value = False
```

Это значит, что мы можем разрабатывать новый код, не открывая сохраненный файл, чтобы увидеть эффект. Но тогда по окончании не забудьте включить функции закрытия документа и скрытия отображения. Если вас это устраивает, создайте новую процедуру:

```
Sub insert_chart (doc as Object, cTitle as String)
End Sub
```

а затем вставьте новую строку в download_stock_price:

```
insert_chart (oDoc, "My Shares")
```

Вам надо поместить ее до выражения save_file_as_ods. Теперь подумаем о том, какой код добавить в новую процедуру.

Если вам случалось вручную создавать график по скачанным данным, то вы уже поняли, что есть проблема: ключ не имеет говорящего имени (он просто пишет Столбец B). Это, конечно, потому, что ключ появляется из строки-шапки – но в нашем случае это тоже данные. Чтобы это исправить, напишем макрос для вставки заголовка:

```
Sub insert_chart (doc as Object, cTitle as String )
Dim oSheet as Object, oCell as Object
oSheet = doc.Sheets(0)
oSheet.getRows.insertByIndex(0,1)
oCell = oSheet.getCellByPosition(1,0)
oCell.String = "Share Value"
End Sub
```

Объясняю, что здесь происходит: мы выбираем первый лист таблицы, затем вставляем новую строку, а затем устанавливаем содержимое B1 в Share Value.

Если вы запустите Main и затем вручную создадите график, то увидите, что столбец значений будет назван Share value. Начало хорошее; теперь пусть макрос создает график сам. Первое, что необходимо выявить макросу, это количество используемых данных:

```
r = 1
```

```
oCell = oSheet.getCellByPosition(0,r)
while oCell.String <> “”
r = r + 1
oCell = oSheet.getCellByPosition(0,r)
wend
```

С помощью этой информации определим диапазон данных в таблице:

```
Dim oRange as Object
Dim oRangeAddress as Object
oRange = oSheet.getCellRangeByPosition(0,0,1,r – 1)
oRangeAddress = oRange.getRangeAddress
```

Теперь мы можем определить данные по x и y в объекте CellRangeAddress:

```
Dim oCellRangeAddress(1) as New com.sun.star.table.
CellRangeAddress
'Set X axis
oCellRangeAddress(0).Sheet = oRangeAddress.Sheet
oCellRangeAddress(0).startColumn = oRangeAddress.
endColumn
oCellRangeAddress(0).endColumn = oRangeAddress.endColumn
oCellRangeAddress(0).startRow = oRangeAddress.startRow
oCellRangeAddress(0).endRow = oRangeAddress.endRow
'Set Y axis
oCellRangeAddress(1).Sheet = oRangeAddress.Sheet
oCellRangeAddress(1).startColumn = oRangeAddress.
startColumn
oCellRangeAddress(1).endColumn = oRangeAddress.
startColumn
oCellRangeAddress(1).startRow = oRangeAddress.startRow
oCellRangeAddress(1).endRow = oRangeAddress.endRow
```

Следующий шаг – определение области для графика. Вы обнаружите, что размер по умолчанию довольно мал, так что сделаем его побольше:

```
Dim oRect as New com.sun.star.awt.Rectangle
oRect.Width= 20000
oRect.Height = 10000
```

Прямо сейчас мы можем написать код, рисующий график:

```
Dim oCharts as Object
oCharts = oSheet.Charts
oCharts.addNewByName(cTitle, oRect, oCellRangeAddress(),TRUE,TRUE)
```

Вы можете запустить макрос, если хотите – и появится график. Однако работа еще не кончена: например, хорошо бы написать название графика и обозначить оси X и Y…

```
Dim oChart as Object
oChart = oCharts.getByName(cTitle).embeddedObject
oChart.HasMainTitle = True
oChart.Title.String = cTitle
oChart.diagram.HasXAxisTitle = True
oChart.diagram.XAxisTitle.String = “Company Symbol”
oChart.diagram.HasYAxisTitle = True
oChart.diagram.YAxisTitle.String = “Closing Value”
```

Теперь у вас есть полностью автоматизированная система для скачивания котировок ваших акций и создания по ним диаграммы. Все, что вам осталось сделать, это включить автоматическое закрытие и скрытие экрана... а на следующее утро встать, выпить кофе и посмотреть на результаты усердной работы.

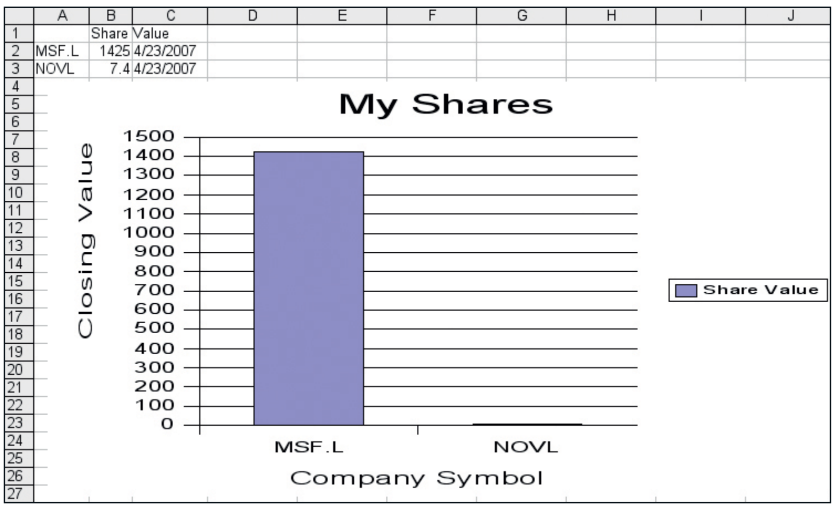
«Погодите!» – слышу я ваш крик, «Мне не нравится диаграмма из столбиков – желаю, чтоб данные отображались в виде пончика; как насчет этого?» Да легко, хоть и не столь очевидно. Мы использовали тип графика по умолчанию, а чтобы использовать любой другой тип, укажите

```
oChart.diagram = oChart.createInstance(“com.sun.star.chart.
DonutDiagram”)
```

Вот полный список доступных типов графиков: AreaDiagram, BarDiagram (по умолчанию), DonutDiagram, LineDiagram, NetDiagram, PieDiagram, StackableDiagram, StockDiagram и XYDiagram.

Теперь можете вытворять что угодно. Можете загрузить CSV-файл прямо в *Calc*, можете создать любой график по данным, и все это – автоматически, без ввода данных. На данном уроке мы занимались Yahoo Finance, так как он свободно доступен и использует CSV-формат. Вы можете, конечно, использовать любой CSV-файл, то есть любой источник данных, при условии, что они загружаются в *OpenOffice.org* – например, приложить этот метод к базам данных. Но если вы примените эти приемы на деле для анализа ваших котировок и урвете крупный куш, уж не забудьте, кто научил вас уму-разуму! LXF

› С минимумом усилий создан красивый график, показывающий, как скверно живется вашим акциям. Победа!



Wireshark: Ловите

Исследуя вашу сеть, можно добыть массу полезной информации. **Грэм Моррисон** применит пакетный анализатор, чтобы продемонстрировать беззащитность ваших данных.





Наш эксперт

Грэм Моррисон
Разработчик свободных программ, фанат KDE и энтузиаст электронной музыки. Он обсуждал синтезаторы не с одним ведущим членом сообщества FOSS.

Не нужно быть специалистом по сетям, чтобы интересоваться их мониторингом. Столь же необязательно родиться крутым хакером. Ясное представление о процессах, происходящих в вашей сети, способствует её максимальной защищенности. Оно даже может подвигнуть вас на полезные действия.

Но для понимания этих процессов, вы должны достаточно близко познакомиться с пакетами, пересылаемыми в вашей сети, чтобы видеть их в упор. Для этой работы есть только один инструмент, и он известен как «сниффер» [«обнюхиватель», ищейка, англ.] пакетов или сетевой анализатор. Такие приложения захватывают необработанные биты и байты потока данных, пока они передаются по вашей сети, и анализируют их для определения используемого сетевого протокола. Затем вы можете «пройти по следу» сетевого соединения и отследить пути взаимодействия между клиентом и сервером.

На этом уроке я собираюсь открыть вам глаза, чтобы вы поняли, насколько в действительности доступны в сети ваши данные. Используя анализатор, мы захватим стандартные пакеты, пересылаемые, когда вы пользуетесь клиентом мгновенных сообщений Gaim для общения с друзьями; затем выделим текст сообщений и убедимся, что он действительно открыт для просмотра всем желающим. Будет полезно хоть чуть-чуть понимать, как работает сеть, но экспертом быть не обязательно. На самом деле, просмотр сетевых пакетов и точный анализ того, что передается по проводам и в эфире вокруг вас – превосходный способ обучения.

Самый популярный анализатор для Linux – инструмент под названием *Wireshark*, но более вероятно, что вы слышали об инструменте под названием *Ethereal*. На самом деле *Wireshark* – это и есть *Ethereal*, а имя поменялось после того, как главный разработчик, Джеральд Комбс [Gerald Combs], сменил работодателя. Его бывшая компания имела права на имя *Ethereal*, но не на код приложения, выпущенный под лицензией GPL. Поэтому все, что Комбс должен был сделать после ухода – это настроить новый репозиторий *Subversion* для своей собственной версии исходного кода и придумать новое имя. В июне 2006 г. *Ethereal* превратился в *Wireshark*.

Для перехвата пакетов вашей сети Wireshark использует системную библиотеку под названием *libpcap*. Если вы используете какой-то экзотический сетевой протокол, не поддерживаемый *libpcap*, то *Wireshark* также не сможет ничего выяснить. Но тем, кто использует специализированные сетевые протоколы, которые даже *libpcap* не понимает, вряд ли понадобится наш урок. Теперь, когда вы знаете всю подоплеку, давайте начнем.

Часть 1: Перехват данных

Важнейшее решение при сетевом анализе – где запускать сниффер. Для личных экспериментов лучше всего сделать это на вашей собственной машине. Если же вы хотите анализировать реальный мир данных вашей сети, запустите *Wireshark* на сервере. Это означает, что любой клиент (не только ваша машина), использующий сервер для передачи данных «в» и «из» Интернет(а), будет виден приложению захвата пакетов. Но ввиду неэтичности этого занятия, нужны веские основания для подобных действий.

После выбора машины для работы настает момент установки *Wireshark* посредством менеджера пакетов. Сниффер включен во множество дистрибутивов по умолчанию и является самодостаточным приложением, использующим GUI для выполнения всех задач – в отличие от многих других инструментов системного администрирования. Хотя вам, может быть, стоит поискать ‘Ethereal’: системы вроде Ubuntu используют это старое имя для обеспечения совместимости пакетов.

Wireshark требует запуска с привилегиями системного администратора – что обычно означает запуск приложения от имени `root`. Если вы беспокоитесь о проблемах безопасности, связанных с запуском такого низкоуровневого инструмента на критичном для системы сервере (чего вы не должны делать в любом случае), хорошим компромиссом может быть запустить *Wireshark* на краткое время, для захвата достаточного количества сетевых данных, и использовать приложение на пользовательском уровне для их анализа.



› **Можно захватывать данные с любого установленного сетевого устройства, включая беспроводные и Ethernet-карты.**



данные в сети

Надзор за устройствами

Вы можете запустить *Wireshark* из командной строки, набрав wireshark. Через пару мгновений появится его GTK 2-интерфейс. На данный момент он больше похож на шестнадцатеричный/двоичный редактор, чем на инструмент анализа сети, просто потому, что пока не содержит данных. Получение данных для приложения – следующий шаг.

Щелкните на первой иконке панели инструментов или выберите Интерфейсы [Interfaces] в меню Захват [Capture]. В открывшемся окне будет приведен список всех имеющихся сетевых устройств. В нем также начнется вывод числа пакетов, проходящих через каждое устройство. Этот список полностью зависит от конфигурации вашей системы. Если ваша машина использует беспроводное подключение к сети, вы, вероятно, увидите устройства **wifi0** или **ath0**; если проводное, то скорее всего они будут озаглавлены **eth0** или, для второй карты, **eth1**. Другие стандартные устройства – **slip0** и **ppp0** для соединений по последовательной линии, используемой в dialup-сетях (помните такие?) [помним, помним. **ppp0** – частый гость на компьютерах современных домашних пользователей, использующих для доступа к провайдеру протоколы PPTP или PPPoE, а также услуги сотовых сетей GPRS/CDMA, – прим. ред.]

Вы также увидите еще два устройства. Одно из них будет обозначено как **lo**, это стандартное кольцевое устройство, реализованное в ядре Linux для тестирования: оно соответствует локальной машине. Другое обозначено как **any** – это мета-устройство, используемое *Wireshark* для объединения всех сетевых подключений к машине. Оно пригодится, если вам необходимо захватывать все, а объем генерируемых данных может оказаться не поддающимся обработке. Намного легче начать с порта, который, как вы точно знаете, используется определенными протоколами, и только потом перейти к захвату данных с других интерфейсов.

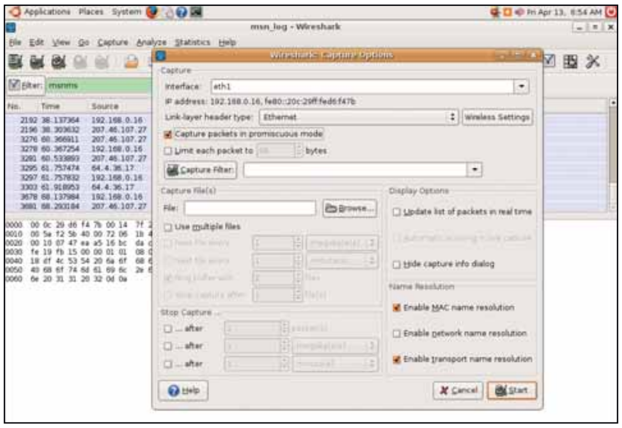
Захват без разбора

Щелчок на кнопке Подготовка [Prepare] рядом с интересующим вас устройством позволит вам произвести тонкую настройку процесса сбора данных. Большинство опций можно оставить как есть, но пару из них вы, вероятно, захотите изменить. Наиболее важный параметр обозначен как Захват пакетов без разбора [Capture Packets In Promiscuous Mode]. Во включенном состоянии он позволяет *Wireshark* перехватывать и сохранять каждый отдельный байт сетевого пакета на самом нижнем уровне. Вне этого режима важная информация о пересылке может быть упущена. Но может ли ваша система работать в нем или нет, зависит от имеющегося драйвера Ethernet, хотя большинство устройств этот режим поддерживают. Другой важный параметр на этой странице назван Обновлять список пакетов в реальном времени [Update List Of Packets In Real Time]. Вероятно, вам не следует пользоваться им при первых попытках применения *Wireshark*, но он очень полезен при быстром анализе. По своей сути он позволяет вам пощупать и просеять сетевые данные без предварительного их захвата – ценно, если вы хотите заметить некоторую специфическую активность в сети.

Внимание, эксперимент!

Удовлетворившись настройками для выбранных вами устройств, нажмите или кнопку Старт [Start] на странице настройки, или кнопку Захват [Capture] в списке интерфейсов. После этого *Wireshark* запишет каждый байт данных, прошедший через ваше сетевое устройство. Чтобы повысить ценность этого упражнения, не мешает сделать через ваше сетевое соединение что-нибудь интересное. Лично я запустил клиента мгновенных сообщений *Gaim* и начал разговор с одним из моих контактов. Есть и другие вещи, которые стоит попробовать, от просмотра нескольких защищенных и не защищенных сайтов (попробуйте купить что-либо в сети!) до передачи/приема электронной почты. Но помните: все, что вы делаете, будет захвачено *Wireshark*. Когда вы сочтете, что накопили достаточно данных для анализа, просто остановите ведение журнала. После этого *Wireshark* загрузит данные в главное окно.

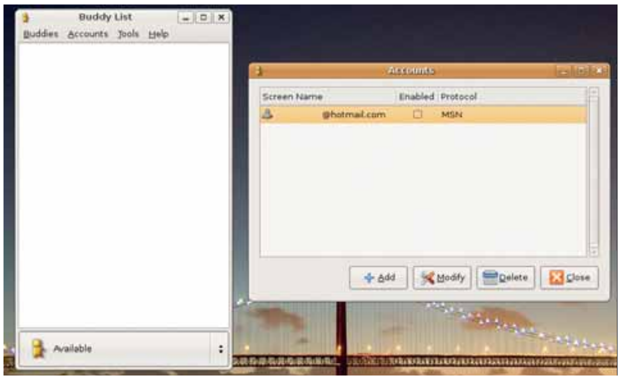
»



› Находясь в режиме захвата без разбора, ваше сетевое устройство «ловит» все пакеты, независимо от адреса отправителя и получателя.

Скорая помощь

Не захватывайте слишком много данных. Помните: *Wireshark* сохраняет все. Если вы загружаете файл в 10 ГБ, *Wireshark* захватит и его.



› Мессенджеры типа Gaim посылают сообщения как простой текст, и любой владелец инструмента типа Wireshark может их читать.

Тактика защиты от вторжений

Простые электронные письма, мгновенные сообщения, путешествие в Web, VNC, IRC, Telnet. Эти протоколы передают данные в простом текстовом формате, и данные легко прочтет любой, кто может запустить *Wireshark* в вашей сети и потратить время на сбор сотен передаваемых пакетов. Вот почему персональные данные, вроде деталей банковского счета и информации о кредитных картах, и даже ваше имя и адрес на самом деле не должны пересылаться через незащищенное интернет-соединение.

По этой причине и возникло множество шифрованных интернет-протоколов. Наиболее общим примером является HTTPS, безопасная версия протокола передачи гипертекста (HTTP), он используется везде: от интернет-банков до сетевых магазинов. То же делает VPN, обеспечивая туннель для удаленной сети через незащищенный уровень (Интернет) к локальному компьютеру – подробности см. на стр. 101 или в LXF93. Любой взломавший соединение увидит разве что зашифрованные пакеты VPN, и не сможет узнать что-либо без взлома ключа шифрования. А это уже совсем другая тема.

Советы по поддержанию безопасности

» Всегда используйте HTTPS для передачи важной информации через Web.

» Используйте SSH для удаленных соединений, желательно с авторизацией по ключу.

» Туннелируйте незащищенные протоколы, вроде VNC, через ваше SSH-соединение.

» Переключитесь на протокол с шифрованием для отсылки и приема почты.

Часть 2: Анализ данных

Сетевые данные – это двоичные данные; вы вряд ли что-либо поймете, загрузив их в текстовый редактор. Сила *Wireshark* в его способности декодировать информационные поля, содержащиеся в захваченных двоичных данных. Путем поиска определенных шаблонов в данных, *Wireshark* может точно определить тип используемого протокола, а по нему извлечь необходимую дополнительную информацию.

Главное окно данных в *Wireshark* (см. ниже) показывает разборку содержимого каждого пакета сетевых данных, которые *Wireshark* умеет распознавать. Первый левый столбец – порядковый номер пакета, в соответствии с его получением. Рядом находится время получения пакета. Справа от времени расположены IP-адреса отправителя и получателя, найденные в пакете – откуда пакет послан и куда направляется. Столбец Протокол [Protocol] отображает аббревиатуру найденного протокола для пакета, а в последнем столбце выводится ценная информация, извлеченная благодаря тому, что *Wireshark* распознал протокол, и какие данные в пакете наиболее полезны.

Фильтр свирепствует

Вся эта информация весьма объемна. Для работы с ней *Wireshark* имеет отличный фильтр, основанный на простом синтаксисе, для уменьшения путаницы на экране; он действительно выделяет пакеты, достойные интереса. Вы можете ввести фильтр напрямую в область поиска над списком пакетов, но для начала проще нажать кнопку Фильтр [Filter] слева от этого поля: здесь имеется несколько стандартных примеров фильтров, наряду с полезным мастером создания и сохранения собственных. Поэкспериментируйте с примерами. Набрав, допустим, ip.addr = = 192.168.0.1, вы получите список только тех пакетов, отправителем или получателем которых является адрес 192.168.0.1 (полезно, если вы хотите проверить отдельного клиента в вашей сети). ip.addr – это поле фильтра, и вы увидите полный список 44 185 возможных вариантов, поддерживаемых *Wireshark*, нажав Справка > Поддерживаемые протоколы [Help > Supported Protocols] и выбрав вкладку Отобразить поля фильтра [Display Filter Fields].

Но пока мы ограничимся одним протоколом, а именно, используемым MSN Messenger для связи с клиентами. Обычно Wireshark определяет используемые протоколы путем поиска по номеру порта в информации пакета, и вы можете увидеть список 820 распознаваемых протоколов, открыв Справка > Поддерживаемые протоколы [Help > Supported Protocols]. Тот, который нас интересует, обозначен как MSNMS – MSN Messenger Service. Можно отфильтровать список пакетов по этому протоколу, набрав в фильтре msnms. Вы заметите, что поле фильтра окрашивается красным, когда невозможно выполнить фильтр, и зеленым, когда основные ключевые слова распознаны.

Введите msnms и нажмите Применить [Apply]; список пакетов обновится, чтобы показать саму MSN-сессию, записанную в пакетах (для обзора выполняемого процесса, см. врезку «Шаг за шагом»). Щелчок на заголовке поля Номер пакета [Packet Number] отсортирует пакеты в порядке их получения, облегчив рассмотрение потока соединения. Первым делом в глаза бросается то, что сама передача представляет собой обмен данных между вашим локальным клиентом и различ-

Скорая помощь

Вы можете сохранить захваченные *Wireshark*'ом данные для дальнейшего анализа. Это полезно, когда вы хотите проследить проблемы сети или в дальнейшем ссылаться на полученные данные.

Изучаем интерфейс Wireshark

Фильтр данных
Используйте любой из сотен поддерживаемых протоколов для тонкой настройки вашего списка пакетов.

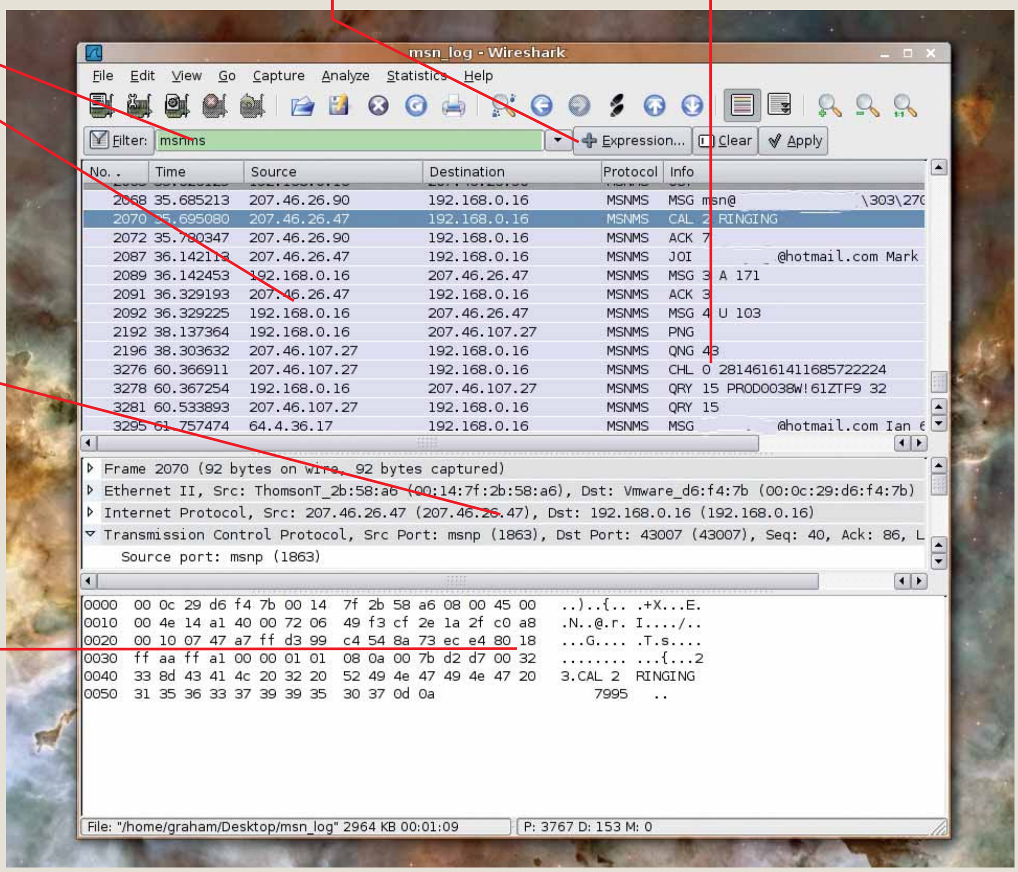
Редактор выражений
Используйте мини-программирование для поиска конкретных деталей в ваших данных.

Информация
Wireshark строит предположения о типе информации.

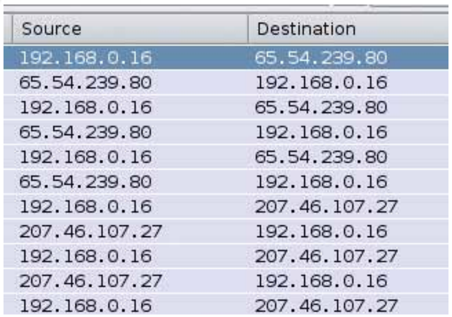
Список пакетов
Каждый захваченный пакет будет помещен в этот список, вместе с IP-адресами отправителя и получателя.

Список заголовков
Общая информация об используемом протоколе для каждого пакета отображается здесь.

Данные пакета
Декодированное содержимое пакета отображается в этой панели.

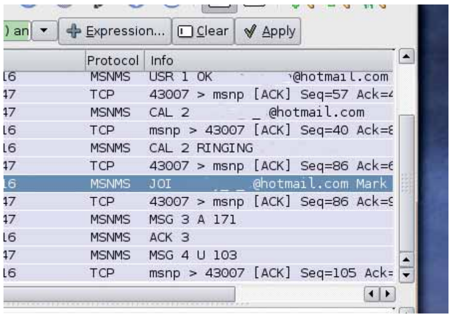


Шаг за шагом: Следим за интернет-беседой



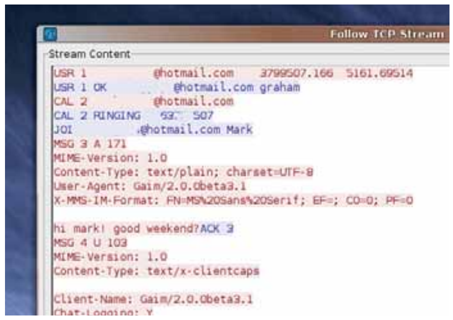
1 Определим сессию

В журнальном интерфейсе *Wireshark*, найдите точку, где клиент и сервер начинают обмен пакетами.



2 Найдем подключение клиента

Следуйте по списку пакетов обмена, пока не найдете пакет, содержащий сигнал 'JOI'.



3 Проглядим пакеты сообщений

Выполните правый щелчок на этом пакете и выберите Follow TCP stream, чтобы увидеть текст беседы.

ными MSN-серверами. Вы можете заметить это по перепрыгиванию вашего IP-адреса из поля Отправитель [Source] в поле Получатель [Destination], поскольку пакеты пересылаются и принимаются между двумя машинами.

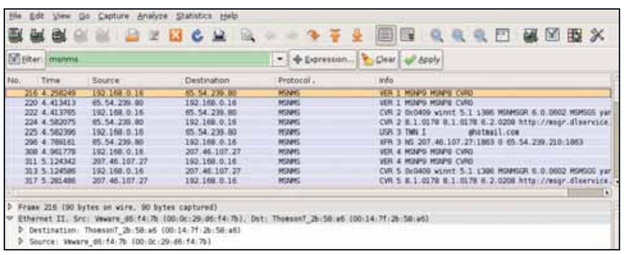
Вы также можете увидеть ход транзакции. Сперва пакеты содержат очень мало информации: первые два просто проверяют версию MSN-протокола, используемую обеими машинами. Первый важный пакет в наших захваченных данных – это третий. Он высылается нашим клиентом и содержит передачу инициализации для MSN-соединения. В него включено имя пользователя, а для большинства MSN-записей это просто адрес на Hotmail. Хорошо уже и то, что не показывается пароль. Это потому, что большинство соединений для аутентификации *MSN Messenger* по умолчанию используют протокол с шифрованием – HTTPS, и эти данные не прочесть как обычный текст. Если же вы используете обычную HTTP-аутентификацию, то ваш пароль будет как на ладони.

Послание в бутылке

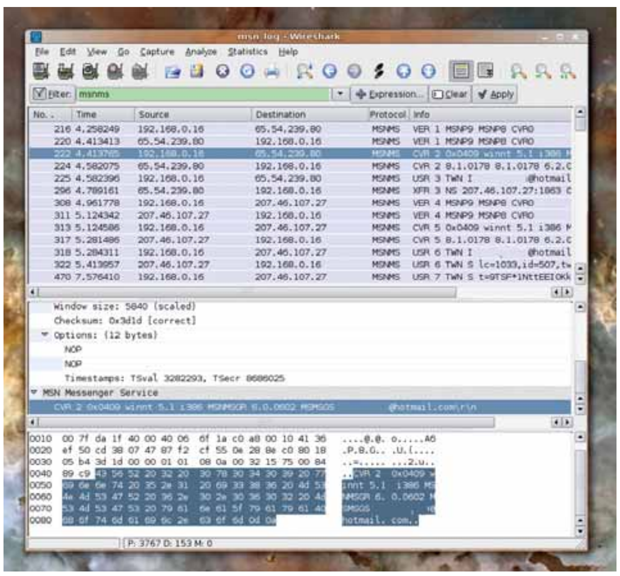
Успешное соединение сопровождается сообщением сервера USR OK, это примерно 15-й пакет в нашей подопытной сессии. Вот где начинаются интересные вещи. Список контактов, появляющийся в нашем клиенте сообщений, хранится на MSN-сервере. Это означает, что для отображения в *Gaim* его нужно переслать вашему клиенту с сервера. Вполне очевидно, что пакеты, посылаемые сервером за сообщением USR OK, будут почти целиком набиты электронными адресами – невероятно ценный ресурс для любого злоумышленника. Все, что ему следует сделать, это часок позаписывать данные в интернет-кафе, и он получит тысячи электронных адресов для своих целей. Мало того: он также сможет кое-что сообразить о связи между адресами, что значительно облегчит успех методов социальной инженерии.

После заполнения списка контактов, следующее действие, которое вы увидите в журнале, это открытие разговора с одним из них. Дамп данных *Wireshark* ясно покажет, что оно произошло, поскольку MSN сервер посылает сообщение RINGING, почти как телефон. Двумя пакетами позднее контакт ответит сообщением JOI. Это говорит о том, что удаленный клиент сообщений присоединился к нашему локальному клиенту для беседы. Сразу после этого мы получаем пакет, содержащий строку X-MMS-IM-Format. Это часть протокола MSN Messenger, помещаемая в начале каждого сообщения, пересылаемого между двумя клиентами. Согласно спецификации протокола MSN Messenger, тело сообщения следует далее в простом текстовом формате – да так оно и есть. Сразу же после заголовка идет частное содержание сообщения. Просмотр ваших интимных разговоров вне контекста и в доступном виде, естественно, заставит вас задуматься о последствиях использования общественных сетей для обмена мгновенными сообщениями. Таким способом вы можете захватить целые беседы, и если вы воспользуетесь захватом пакетов вместе со взломанной точкой беспроводного доступа, то сможете незаметно войти в курс дел всех своих соседей.

Надеюсь, весь этот процесс напугал вас достаточно, чтобы включить наконец для своей точки доступа WPA-шифрование (вы же знаете, что WEP небезопасен, так ведь?) и, может быть, даже использовать зашифрованные каналы связи при необходимости пользования открытой сетью. Однако анализ пакетов – штука захватывающая, а иногда и необходимая для проверки совместимости версий в таких протоколах, как Samba. LXF



› Можно просмотреть заголовок каждого захваченного пакета, сдвигая вниз разделитель панелей над списком пакетов.



› Конечный результат проекта – умение подслушивать IM-беседы: горький опыт, способный заставить вас усилить безопасность.

Mono: Работаем

Два ядра позволяют сделать больше – по крайней мере, так уверяет отдел маркетинга Intel. Проверим это вместе с **Полом Хадсоном.**





Наш эксперт

Пол Хадсон полагает, что Mono – лучшая вещь со времен мультфильма Pinky and the Brain, и сейчас поддерживает два проекта на основе Mono на SourceForge.

Дивлюсь я на мою жену. Я при переходе улицы с трудом успеваю поглядеть в обе стороны, а она способна гладить, говорить по телефону и смотреть телевизор одновременно. Она правда думает обо всем сразу, или ее мозг мгновенно переключается с задачи на задачу?

Долгое время компьютеры были ограничены только последним вариантом. На вашей системе Linux одновременно работает около 100 программ. Вам видны лишь некоторые из них, вроде *X* или *Nautilus*, но есть еще и другие – апплет громкости, *syslogd*, *Metacity*, *D-BUS*, *Cron* и так далее. Большую часть времени они бездействуют в фоновом режиме, но когда два или более вступают в дело одновременно, ваш процессор начинает ими жонглировать. Обычный стандартный процессор без включенного Hyperthreading может выполнять только один процесс в заданный момент времени.

Чтобы избежать подвисания при запуске *OpenOffice*, каждый процесс получает период времени – доли секунды, обычно менее 100 мс – на выполнение кода. По истечении этого времени процесс приостанавливается, и свой квант времени получает другая программа. Если квант равен 100 мс, то за секунду успевают поработать десять различных программ; человеку за этим не уследить, и ему кажется, что все они работают одновременно.

Так продолжалось много лет; но на новых двух- и многоядерных чипах от AMD и Intel или любой старой SMP-системе с двумя физическими одноядерными процессорами все по-другому. Эти устройства могут действительно исполнять множество процессов сразу, благодаря наличию нескольких чипов: двухъядерный чип может выполнять два процесса одновременно, а четырехъядерный – четыре. Внутри процесса работают «потоки», которые представляют собой отдельные задачи внутри программы, способные работать параллельно с другими задачами. Однако даже самый красивый и изящный в мире код на C#, содержащий только один поток, использует всего четверть от четырехъядерной мощи.

На нашем уроке вы изучите, как создавать потоки в Mono, для запуска приложения одновременно на нескольких ядрах. Чтобы сделать тему более захватывающей, создадим «взломщика» хэшей SHA1. SHA1 – это алгоритм хэширования, спроектированный для создания 40-символьной уникальной последовательности битов из входного текста. Хэши обычно используются для проверки целостности информации – если вы скачаете 4-ГБ образ DVD, у которого искажен 1 КБ информации, то полученный хэш SHA1 будет совершенно отличаться от исходного. SHA1 и другие функции часто используются для хранения паролей, так как исходное значение пароля по хэшу не восстановить – хотя можно генерировать SHA1-ключи для всех возможных строк, чтобы увидеть совпадения. Но сначала займемся чем-нибудь попроще.

Попасть в квадрат

Первым нашим проектом этого урока будет возведение в квадрат 1000 чисел. Мы начали с такого примера, потому что его очень легко распараллелить: не требуется обмена данных между потоками. Создайте новое консольное C# приложение в MonoDevelop, назовите его *Hackaday* и поместите следующую строчку вверху его cs-файла:

```
using System.Threading;
```

Магическая строка using позволит нам использовать потоки. Нам также потребуется 4 переменных: одна будет отслеживать, сколько чисел надо создавать, другая будет отвечать за количество потоков, третья установит, сколько чисел генерировать на поток, а четвертая будет хранить генератор случайных чисел. Без первых трех переменных на самом деле можно обойтись, записав их как константы, но потом с ними уже не поиграешь!

Итак, добавьте четыре переменных до определения метода static void Main():

```
static int NumsToGenerate = 1000;
static int NumThreads = 4;
static int NumsPerThread;
static Random Rand = new Random();
```

Начинается настоящее дело: создание потоков. Каждый созданный поток будет выполнять метод, который определим мы. Метод может быть каким угодно, принимать любые параметры и даже вызывать другие методы. Но пока будем проще: пусть каждый поток пробегает в цикле от 0 до NumPerThread, генерирует число от 1 до 1000, затем возводит в квадрат и выдает результат.

Вот этот метод:

```
static void DoFunk() {
 for (int i = 0; i < NumsPerThread; ++i) {
  int num = Rand.Next(1, 1000);
  Console.WriteLine("Thread #{0} says: {1} squared is {2}",
  Thread.CurrentThread.Name, num, num * num);
```

» **Месяц назад** Ради практики в ООП мы создали клон *Civilization*.

С потоками

```
  }
}
```

DoFunk() – не очень конкретное имя, но так как мы будем использовать его во многих программах, сойдет и оно! Основная идея в том, что каждый их четырех потоков будет прокручивать 250 случайных чисел и выдавать квадрат каждого из них. Каждый поток будет ссылаться сам на себя с помощью Thread.CurrentThread, и в этом случае мы считываем Name – строку, назначаемую каждому потоку для упрощения отладки.

Синтаксис {0}, {1}, {2} – просто быстрый способ написать сложные вызовы WriteLine() за один раз: Mono автоматически подставляет параметры, то есть замещает {0} на Thread.CurrentThread.Name, {1} на num, {2} на результат num*num.

Остается только метод Main(), которому надо вычислить, сколько чисел должен обработать каждый поток, затем создать потоки и запустить их. При создании каждого потока в его конструктор передается имя метода, который мы хотим запустить. Вы все поймете, взглянув на код – вот он:

```
static void Main(string[] args) {
  NumsPerThread = NumsToGenerate / NumThreads;
  for (int i = 0; i < NumThreads; ++i) {
    Thread thread = new Thread(DoFunk);
    thread.Name = Convert.ToString(i);
    thread.Start();
  }
}
```

Итак, считая от 0 до 4, создадим поток и велим ему запустить метод DoFunk(), а назовем его по номеру итерации, на которой он создается. Хотя все потоки будут созданы, ни один их них не запустится до тех пор, пока не будет вызван метод thread.Start(), после которого они начнут выполнять методы DoFunk(). Нажмите F5, чтобы собрать и запустить программу, и увидите вихрь чисел в окне вывода результатов.

Разделяй и... разделяй

Вы заметите четыре важных момента в работе программы:

1 Каждый поток имеет доступ к генератору случайных чисел Rand и NumThreads, потому что они помечены как 'static', то есть каждый поток может читать и писать их.

2 Каждый поток создает собственные случайные числа. Это потому, что переменная num объявлена локально в каждом потоке, поэтому у них есть по копии этой переменной, чтобы ей управлять.

3 При выводе программы вы заметите, что потоки не выводят каждый по строке, типа 012301230123. Более вероятно, что сначала поток 0 напечатает десять строк, затем поток 1 напечатает 10 строк, и так далее, то есть 000000000011111111112222222222333333333.

4 Программа *hackaday.exe* ждет, пока все потоки не закончат свою работу.

Пункт 1 показывает, что потоки могут иметь общие переменные. В этом разница между процессами и потоками: порождаемые процессы независимы, а потоки разделяют большинство своих данных. Исключения составляют переменные, объявленные локально, как, например, num. Пункт 3 иллюстрирует то, что говорилось о квантах времени выше: каждый поток получает свой квант и исчерпывает его, чтобы передать работу следующему потоку.

Пункт 4 возник потому, что по умолчанию .NET создает не фоновые (foreground) потоки и не позволит завершить программу, пока они не отработают. Фоновые потоки, напротив, автоматически заканчивают работу, когда завершается родительский процесс. Попробуйте перед thread.Start() набрать thread.isBackground=true; затем перезапустите программу. На этот раз программа завершится быстрее: создав все потоки, Main() завершится, и потоки автоматически ликвидируются.



Сконцентрируемся на пункте 1, так как вопрос разделения данных – один из самых сложных. На техноязе то, чем мы занимаемся, называется потокобезопасность, и означает, что ваше приложение не сломается, если два потока попытаются сделать одно и то же в один момент. Что если два потока вдвоем примутся читать статическую переменную? Чтение переменных менее проблемно, но тоже небезопасно: легко нарваться на «состояние гонки» [race condition]. Не буду объяснять, что это такое, сейчас: из кода все станет ясно.

Для начала попробуем безопасным образом писать в переменные из потока. Наш старый код генерировал случайные числа для возведения в квадрат, но сейчас мы собираемся создать список из целых чисел (об этом см. LXF92), и каждый поток будет считывать первый элемент

> **«Вот это и называется гонкой: потоки стараются опередить друг друга, и результат непредсказуем»**

из списка, удалять его и затем возводить в квадрат. Нам не нужна ситуация, когда все четыре потока прочитают первый элемент, затем поток 0 удалит его, поток 1 примется удалять следующий элемент, поток 2 – еще один, а поток 3 – еще один, и выйдет, что мы сосчитали квадрат для первого числа 4 раза, уничтожили 2-й, 3-й и 4-й элементы, сосчитали квадрат для 5-го элемента... и так далее.

C# позволяет легко разрешить эту проблему с помощью выражения lock, отмечающего критические секции кода. Внутри критического блока в заданный момент времени может находится только один поток. Любой другой поток, дойдя до lock-секции, будет ждать, пока первый поток не выйдет из нее. Отсюда следует, что нам надо блокировать любые общие переменные, прежде чем изменять их, чтобы предотвратить двойные изменения. »

Вот так параллельность!

Мы здесь обсуждаем так называемые «ошеломляюще параллельные» алгоритмы, которые хорошо распределяются по процессорам, так как каждая операция абсолютно не зависит от остальных. Как легко представить, не много задач попадает в эту категорию: физика частиц – да, фракталы – да, и несколько других классов. А вот со сжатием видео уже не все просто, потому что большинство кодеков кодируют изменения с предыдущего кадра, и вы не можете сжать кадр до того, как был обработан предыдущий. Использование ключевых кадров смягчает проблему, но есть задачи – в основном криптография, поблочное шифрование с обратной связью – которые нипочем не распараллелить.

И все-таки не беспокойтесь, если ваше приложение не может быть распараллелено на 100%. Если вы создали шахматную программу, которая выполняет в одном потоке все, кроме ИИ компьютера, который в фоновом потоке будет искать наилучший ход, это уже неплохо.

» Блокировка потоков

Так как мы будем использовать коллекцию List, надо добавить новое выражение в секцию using:

```
using System.Collections.Generic;
```

Удалите все ранее определенные переменные и вставьте следующее:

```
static List<int> Numbers = new List<int>();
```

Метод Main() должен подготовить 2000 чисел для Списка. Так как мы удалили все переменные, число потоков будет зашито в программу в виде константы. Вот новый код Main():

```
public static void Main(string[] args) {
  for (int i = 0; i < 2000; ++i) {
    Numbers.Add(i);
  }
  for (int i = 0; i < 4; ++i) {
    Thread thread = new Thread(DoFunk);
    thread.Name = Convert.ToString(i);
    thread.Start();
  }
}
```

Серьезная работа возложена на метод DoFunk(): ему надо вытащить число из списка и возвести его в квадрат, блокировав при этом список Numbers, чтоб не вмешались другие потоки. Вот как выглядит код DoFunk():

```
while (Numbers.Count > 0) {
  lock (Numbers) {
    int Num = Numbers[0];
    Numbers.RemoveAt(0);
    Console.WriteLine("Thread #{0} says: {1} squared is {2}",
    Thread.CurrentThread.Name, Num, Num * Num);
  }
}
```

Цикл будет выполняться, пока в Numbers остаются числа. Но первым делом надо заблокировать список – lock(Numbers). Первый поток, который доберется до этого кода, обнаружит, что Numbers свободен, и заблокирует его. Другие потоки обнаружат, что Numbers заблокирован первым потоком, и не пойдут дальше, пока первый поток не снимет блокировку. Первый поток прочтет первое число, удалит его из списка, возведет в квадрат и выведет на экран, затем, достигнув конца блока lock, освободит Numbers. Второй поток обнаружит, что Numbers свободен, заблокирует его, выполнит свою работу, и так далее. Попробуйте запустить и посмотреть, что случится.

Все выглядит замечательно до тех пор, пока вы не дойдете до конца, когда вы увидите следующее:

```
Unhandled Exception: System.ArgumentOutOfRangeException:
Argument is out of range.
Parameter name: index
at System.Collections.Generic.List`1[System.Int32].get_Item
(Int32) [0x00000]
at hackaday.MainClass.DoFunk () [0x00000]
at (wrapper delegate-invoke) System.MulticastDelegate:invoke_
void ()
```

Китайская грамота, да? Что ж, так нам пытаются сообщить, что возникло «состояние гонки». То есть два потока (или более) параллельно пытаются достичь одного результата, и мы из-за непредсказуемой работы планировщика виртуальной машины получаем неожиданные результаты. Взгляните на код – в частности, на метод DoFunc(). Проблема находится в строке int Num = Numbers[0] – там, где вызывается внутренний метод Mono get_item(). Посмотрим, что здесь может вызвать проблему.

Нашли? Если нет, давайте я покажу, что происходит в случае двух потоков исполнения:

1. **Поток 1**: Numbers > 0.
2. **Поток 1**: Numbers свободен. Блокируем его.
3. **Поток 2**: Numbers > 0.
4. **Поток 2**: Numbers блокирован. Ждем.
5. **Поток 1**: Взять первое число, удалить, возвести в квадрат и вывести на экран.
6. **Поток 1**: разблокировать Numbers.
7. **Поток 2**: Numbers свободен. Блокировать его.
8. **Поток 1**: Numbers > 0.
9. **Поток 1**: Numbers блокирован. Ждем.
10. **Поток 2**: Взять первое число, удалить, возвести в квадрат и вывести на экран.
11. **Поток 2**: Разблокировать Numbers.

... и так далее. Но в конечном счете произойдет следующее:

1. Поток 1: Numbers свободен. Блокируем его.
2. **Поток 2**: Numbers > 0.
3. **Поток 2**: Numbers блокирован. Ждем.
4. **Поток 1**: Взять первое число, удалить, возвести в квадрат и вывести на экран.
5. **Поток 1**: Разблокировать Numbers.
6. **Поток 2**: Numbers свободен. Блокируем его.
7. **Поток 1**: Numbers не больше 0. Закончили.
8. **Поток 2**: Взять первое число... БАЦ!

В этой последовательности, Numbers начинает с ровно одним оставшимся элементом, и **поток 1** его блокирует. В то же время **поток 2** проверяет Numbers на наличие элементов, и его ждет успех – так как Numbers все еще содержит число. Но заблокировать Numbers он не может, поэтому ждет. **Поток 1** продолжает работу, удаляет первый элемент и возводит его в квадрат, затем разблокирует Numbers. **Поток 2** теперь блокирует Numbers, думая, что там еще остались числа, и пытается считать оттуда, затем отбрасывается исключение 'System.ArgumentOutOfRangeException', так как Numbers[0] не существует.

Если запустить программу несколько тысяч раз, обнаружится, что она не всегда завершается аварийно. Полный произвол: **поток 1** может заблокировать Numbers и удалить последнее значение до того, как **поток 2** сможет прочитать оставшееся число элементов. Вот почему это зовется ситуацией гонки: потоки забегают один за другой, и между проверкой значения и его использованием может случиться все что угодно.

Решение проблемы состоит в том, чтобы проверить значение Numbers сразу после получения блокировки, чтобы учесть случай отсутствия чисел. Добавьте следующую строку сразу после вызова lock:

```
if (Numbers.Count == 0) break;
```

Теперь запуск программы увенчается успехом, так как мы устранили ситуацию гонки.

В белых перчатках

Пришла пора для главного проекта месяца: взлома SHA1 методом грубой силы. Породим 26 потоков (по одному на каждую букву алфавита), затем заставим их считать SHA1-хэш для всех возможных слов, начинающихся с этой буквы. Программа прочтет из командной строки максимальный размер слова и хэш SHA1, и остановится, как только

Скорая помощь

Имя потоку дается только один раз, поэтому сразу выберите что-то значимое.

Ваше домашнее задание

Вы заметили, что я предоставил вам отнюдь не оптимальный алгоритм для взлома SHA1, и не случайно: это Linux Format, а не Black Hat Format. Однако для вас будет интересным упражнением проверка времени, которого потребует ваша атака на SHA1. Три совета начинающим:

1. Ширина охвата ('a','aa','ab','ac') лучше, чем глубина ('a','aa','aaaa','aaab').
2. Изменение одной строки может сократить поиск 5-символьного слова с 22 секунд до 0.4 секунды.
3. Много потоков может навредить производительности – разве что у вас куча процессоров.
4. Постоянное выделение переменных – пустая трата времени.

Достигнув хороших результатов, поместите свой код на форум Linux Format, чтобы другие смогли оценить вашу работу (www.linuxforum.ru). Дерзайте!

Представляем SHA2

SHA1 – не особенно сильный алгоритм хэширования, и он уже не рекомендуется экспертами по безопасности. Однако C# предлагает поддержку для более мощных хэш-функций, включая семейство SHA2 с невероятно сильным SHA512. Конечно, имейте в виду, что наша «лобовая атака» годится только для паролей, так как у большинства пользователей пароли короче 8 символов. На попытку найти совпадение для более длинного сообщения, например, электронного письма, потребовались бы годы.

найдет слово, соответствующее хэшу. Максимальный размер слова будет использоваться для ограничения пробных входов: например, если максимум установить на 4, то поток проверит сначала 'aa', 'aaa', 'aaaa', 'aaab', 'aaac'... 'aaay', 'aaaz', 'aaba' и так далее до 'azzz'.

Здесь не хватит места для всей программы, потому что она занимает около 90 строк (вы найдете ее на диске). Но прежде чем браться за дело, обсудим несколько важных моментов:

1 Программа ожидает на входе 2 параметра: максимальный размер создаваемого слова и хэш, которому оно должно соответствовать. Если параметры не предоставлены, программа завершается. Заметим, что метод DoFunk() – точнее, любой метод, который поток выполняет при своем запуске – должен не иметь параметров, либо принимать только один параметр. Если надо передать несколько параметров, следует создать класс, содержащий все необходимые данные, а затем передать объект этого класса в поток.

2 Main() создает 26 потоков, по одному на каждую букву алфавита. Это делается следующим образом: буква 'a' преобразуется к целому значению в кодировке ASCII, затем добавляется число i и опять преобразуется в букву. I находится в диапазоне от 0 до 25, что дает нам 'a','b','c' и т.д. Затем оно передается в thread.Start().

3 Созданные потоки помещаются в список с незатейливым именем Threads.

4 Причина, по которой мы отслеживаем потоки – мы хотим, чтобы программа дождалась всех потоков, а потом уж продолжила свою работу. Для этого и метод thread.Join(): он ждет завершения потока.

5 Это, в свою очередь, сделано ради подсчета, сколько времени потребуется для завершения работы всех потоков. Если мы не сделаем Join() для каждого потока, Main() продолжит выполнение и спутает нам хронометраж.

6 Подсчет хэша SHA1() на .NET – штука мучительная, и я решил эту проблему, написав вам в помощь небольшой метод. Чтобы посчитать SHA1-хэш любого слова, просто отправьте ее в метод Sha1().

Откинув все это в сторону, вы увидите, что метод DoFunk() поразительно мал. На самом деле он только принимает параметр (начальную букву), а затем вызывает другой метод. Но обратите внимание на то, как он принимает параметр типа object, затем преобразует его (char) в символ. Это неизбежность в .NET, но ее легко обойти.

DoFunk() вызывает NextLetter(), передает ей начальную букву и номер 2. Второй параметр – это уровень, то есть размер слова.

Если мы велим программе искать слова максимальной длины 6, то цикл пойдет, обнаружит, что уровень (2) меньше или равен максимуму (6) и войдет в цикл от 0 до 25, а затем перезапустит себя. Вот как будет выглядеть поток вычислений после вызова NextLetter("a",2):

```
if (2 <= 6) {
 for each alphabet letter {
  string = "a" + letter;
  hash = Sha1(string);
  if (hash is what we're looking for) {
   print happy message
  } else {
   print sad message
  }
  NextLetter(string, 3);
}
```

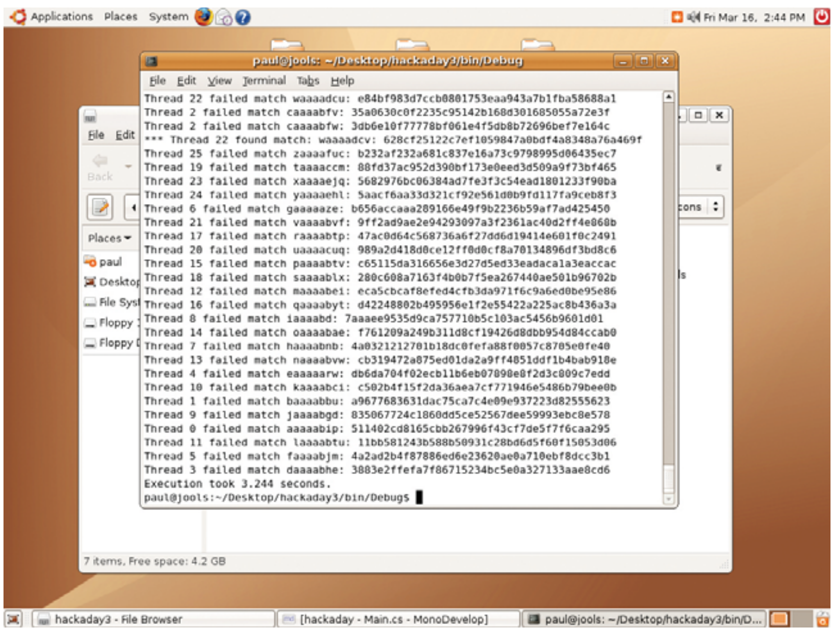
О да, рекурсивная функция может здорово надсадить мозги, но самый простой способ понять ее работу – рассовать повсюду выводы сообщений Console.WriteLine(), чтобы проследить внутреннюю логику.

Если вы поняли, как генерируются входные слова, дальше все просто. Программа использует потоки, и так как потоков 26, она будет плавно рассредотачиваться при добавлении процессоров – даже четырехъядерную машинку обеспечит работой под завязку! Вы можете видеть результат на экранном снимке внизу. Заметьте, как другие 24 потока выводят сообщения, найдя соответствие. Это в чистом виде гонка.

Короче, многопоточность не так уж сложна – по крайней мере, если вы разобрались с блокировками, гонками, присоединениями и фоновыми потоками. Верно? Согласны? LXF

Скорая помощь

Помните, что блокировки заставляют потоки простаивать, так как только один поток может находиться в критической секции. Чем меньше блокировок, тем быстрее будет работать ваш код.



› Наш готовый проект считает SHA1-ключи для всевозможных комбинаций букв и ищет соответствие. Обратите внимание, как 24 других потока выводят сообщения, когда таковое найдено.

» **Через месяц** Упростим доступ к базе: скрестим SQL и C#.

Canvas: Холст

Новый HTML-элемент canvas позволяет программировать графику в браузере. **Дэн Фрост** покажет вам, как использовать этот мощный тэг.





Наш эксперт

Ден Фрост
Дэн работает в Брайтоне техническим директором 3ev, агентства по разработке web-приложений. Он работал на проекте CMS TYPO3, а в настоящий момент занимается CMS-приложением 'Involve' для 3ev.

Рисование графики на web-страницах обычно отдавалось на откуп *Flash* или библиотекам на стороне сервера. Больше такому не бывать! Забудьте зависть и удовлетворите свои амбиции стать художником – по крайней мере в *Firefox* и *Opera*, благодаря тэгу <canvas>.

Впервые введенный фирмой Apple, этот тэг предоставляет программисту холст, чтобы рисовать на нем прямые, дуги, квадраты и так далее. Комбинируя простые элементы, можно получать блестящие результаты – как, вы узнаете чуть позже. В настоящий момент тэг canvas поддерживается браузерами *Firefox*, *Opera*, *Safari* и включен в спецификацию HTML 5 группой WHATWG (Web Hypertext Application Technology Working Group – Рабочая группа по технологии гипертекстовых приложений), что вселяет надежду на его грядущую повсеместную реализацию. В качестве краткого введения, на данном уроке я собираюсь показать, как создать простые фигуры с помощью тэга canvas, поместить эти фигуры в объектную модель и, наконец, сделать их интерактивными.

Привет, мир графики

Чтобы работать с тэгом canvas, достаточно создать небольшой HTML-файл и открыть его в *Firefox*. Для начала рассмотрим несложный пример. Тэг canvas добавляется на страницу обычным образом. Атрибуты «ширина» и «высота» говорят сами за себя; но вы также можете включить стандартные HTML-атрибуты, типа class, id, style и других:

```
<canvas id="canvas" style="border : thin solid black;"
width="600" height="400"></canvas>
```

Добавьте этот кусок кода на пустую HTML-страницу и откройте ее в браузере с помощью File > Open (Файл > Открыть). Все, что вы видите – это пустая область с границей, так как наш код устанавливает тэг canvas, но ничего другого не делает: просто обозначает пустой холст на странице.

Самое интересное начинается в JavaScript. JS используется для рисования линий, кривых, областей и изображений на объекте canvas. Чтобы создать две линии, добавьте это в только что созданный файл и обновите страницу в *Firefox*:

```
<canvas id="canvas" width="600" height="400"
style="background-color : #c0c0c0;"></canvas>
<script>
var canvas = document.getElementById('canvas');
ctx = canvas.getContext('2d');
ctx.lineTo(100, 100);
ctx.lineTo(200, 300);
ctx.lineTo(300, 100);
ctx.lineTo(300, 300);
ctx.stroke();
</script>
```

Давайте разберемся. Сначала мы получаем HTML-элемент (или узел) по id 'canvas'. Затем из элемента canvas мы получаем контекстный объект, его мы рассмотрим далее. Объект используется для собственно рисования.

Вот и все, что необходимо для применения тэга canvas: HTML-тэг и немного JavaScript. Вся ваша разработка может вестись с помощью *Firefox* и тестироваться вне сервера, так как логика и рисование находится на стороне клиента. Но что мы можем? Чтобы подстрекнуть свое воображение, гляньте на панель *Dashboard* от Apple (сайт **www.apple.com/macosx/features/dashboard**). Кроме того, всплывает масса мелких примеров: от реализации *Paint* (*CanvasPaint*) до восхитительных игр вроде *Blobsallad* (**http://blobsallad.se**) и *Canvascope* (**www.abrahamjoffe.com.au/ben/canvascape**).

Программировать с помощью canvas совсем не сложно – мой пример это доказывает. Но если вам неохота писать тысячи строк лапшеобразного кода для отрисовки сложного извива ДНК, поможет объектная модель. Наша следующая задача – поместить функциональность canvas в очень простой (50 строк) модуль, для упрощения управления фигурами.

Расширяем холст

Спецификация на canvas определяет множество функций для рисования простых фигур и линий, вроде прямоугольников, дуг и кривых второго порядка, которые можно комбинировать для создания более сложных, изощренных фигур. По мере того, как я буду описывать различные функции и возможности тэга, копируйте их в ранее созданный файл, чтобы понять смысл функций и как их использовать.

Фактически мы рисуем не на объекте canvas, а на его 2D-контексте. Чтобы получить контекст canvas, мы используем функцию getContext. Контекст – это объект, содержащий функции для рисования линий, блоков цветных изображений и так далее. Линии он называет штрихами [stroke], а цветовые блоки – заливкой [fill]:

» **Месяц назад** Мы использовали *Blender* и Python для создания каркаса чата.

для web-картин

```
var canvas = document.getElementById('canvas');
ctx = canvas.getContext("2d");
ctx.strokeRect(50, 50, 100, 100);
ctx.stroke();
ctx.fillRect(100, 100, 100, 100);
ctx.fill();
```

Большая часть функций понятна: strokeRect() рисует прямоугольник, fillRect() его заполняет. У strokeRect() и fillRect() похожие параметры – x и y верхнего левого угла прямоугольника, а также ширина и высота. Canvas ведет отсчет от левого верхнего угла, то есть 50, 50 означает 50 пикселей слева и 50 пикселей сверху. Тем, кто привык мыслить в терминах координат обычных графиков, тут легко и запутаться! Заметим также, что нам надо вызывать функции .stroke() и .fill(). Не забудьте это сделать, иначе останетесь с пустым холстом.

Цвет линий и заливки устанавливается на холсте с помощью lineStyle и fillStyle, которые принимают похожие аргументы. Самый простой способ ввести цвет – указать его: ctx.lineStyle='red' – или обозначить шестнадцатеричным числом: ctx.lineStyle='#a0b0c0';. Для получения прозрачности предусмотрен альфа-канал в функции rgba(), принимающей четыре параметра: красный, зеленый, синий и альфа. Например:

```
ctx.lineStyle = 'red';
ctx.fillStyle = 'rgb(200, 100, 0, 0.5)';
```

Когда две линии пересекаются, стиль их сочленения можно выбрать с помощью lineJoin:

```
ctx.lineJoin = 'curve';
```

Две последние важные концепции – перенос и поворот. Перенос – это передвижение в другую точку холста; поворот, очевидно, и есть поворот холста. Оба действия производятся до того, как вы начнете рисовать. Например, если вы хотите нарисовать прямоугольник под 45°, то сначала вам надо добавить функцию поворота:

```
ctx.rotate(-45 * Math.PI / 180);
ctx.strokeRect(50, 50, 100, 100);
ctx.stroke();
```

Все эти настройки внутри холста, то есть lineStyle, fillStyle, поворот и перенос, можно сохранять в стеке состояний, а потом брать их оттуда. Это позволяет понаделать кучу стилевых настроек, а затем разом их отменить, закончив рисование:

```
ctx.save(); // Сохранить текущее состояние, чтобы мы могли его
восстановить
ctx.rotate(-45 * Math.PI / 180);
ctx.lineStyle = 'blue';
ctx.strokeRect(50, 50, 100, 100); // нарисовать прямоугольник
ctx.stroke();
ctx.restore(); // восстановить lineStyle и поворот
```

Функции save и restore очень полезны, если вы производите много переносов и поворотов: они избавят вас от необходимости держать в памяти всю серию изменений.

Рисование фигур

Чтобы создать коллекцию фигур-шаблонов, заготовим пару полезных объектов: обертку холста и обертку фигуры. Обертка фигуры представляет наибольший интерес – она инкапсулирует различные сложные фигуры. Обертка холста будет следить за размещением фигур на холсте.

```
function draw() {
  ic = new iCanvas('canvas');
```



› Нарисовав объект, превратите его шаблон для многократного использования.

```
  // добавьте здесь все свои фигуры
  setupInteraction();
}
function setupInteraction() { /* мы рассмотрим ее позднее*/ }
  iCanvas = {};
  iCanvas = Class.create();
  iCanvas.prototype = {
    /* ..вырезано.. */
  }
  // используется для инкапсуляции фигур
  iShape = {};
  iShape = Class.create();
  iShape.prototype = {
    /** ..вырезано... **/
}
// рисование будет проходить в этой функции
renderShape : function() { },
}
```

Полная версия исходного кода доступна по адресу www.linuxformat.co.uk/mag/canvas.htm, но единственной важной частью является renderShape(), так как все, что мы будем туда писать, будет родным кодом холста – другие функции (например, drawShape()) добавляются для упрощения кодирования.

JavaScript сопровождается HTML-страницей, которая вызывает функцию draw() в момент загрузки. Следующий пример немного прояснит ситуацию. Нарисуем простой прямоугольник – скопируйте следующий код в функцию draw() выше, где написано добавьте здесь все свои фигуры. Он создает экземпляр класса shape и использует функцию strokeRect(). Для рисования фигуры вызывается функция drawShape обертки холста:

```
box = new iShape();
box.renderShape = function() {
  // this.c это контекст холста
  this.c.fillRect(0, 0, 100, 100);
  this.c.strokeRect(0, 0, 100, 100);
  this.c.stroke();
}
ic = new iCanvas('canvas');
// теперь рисуем фигуры
ic.drawShape(box, {x : 200, y : 100});
ic.drawShape(box, {x : 100, y : 100});
ic.drawShape(box, {x : 250, y : 150});
```

»

» Я вызывал drawShape несколько раз, чтобы показать, как можно повторно использовать фигуру после ее создания. Просто передайте объект и некоторые новые ссылки, и готово – получена новая копия фигуры.

Все, что делается с тэгом canvas, можно, по большей части, заложить в объект фигуры и повторно использовать его на холсте. Теперь нарисуем более сложный пример: дачный домик. Это всего лишь набор прямоугольников и треугольников:

```
house = new iShape();
house.renderShape = function() {
  this.c.fillStyle = 'white';
  this.c.fillRect(0, 0, 100, 100); // ставим белый фон
  this.c.strokeRect(0, 0, 100, 100);
  this.c.fill();
  this.c.strokeRect(20, 60, 20, 40); // дверь
  // окна
  this.c.fillStyle = "rgba(0, 0, 250, 0.3)";
  this.c.fillRect(60, 60, 20, 20);
  this.c.fillRect(20, 20, 20, 20);
  this.c.fillRect(60, 20, 20, 20);
  this.c.fill();
  this.c.fillStyle = "rgba(250, 0, 0, 0.3)";
  this.c.moveTo(0,0);
  this.c.lineTo(50,-30);
  this.c.lineTo(100,0);
  this.c.fill();
  this.c.stroke();
}
```

fillStyle использует функцию rgba(), позволяющую нам установить прозрачность. Вы можете установить прозрачность «для всего» с помощью .globalAlpha=0.5;. Часть lineTo рисует отрезок на холсте [от текущего положения] до точки (x,y). В данном примере мы используем ее для рисования крыши.

Как и в предыдущем примере, мы можем повторно использовать объекты на холсте с помощью функции drawShape(). Написав

```
ic.drawShape(house, {x : 200, y : 100});
ic.drawShape(house, {x : 300, y : 100});
ic.drawShape(house, {x : 400, y : 100});
```

вы отстроите целую уличку.

Скорая помощь

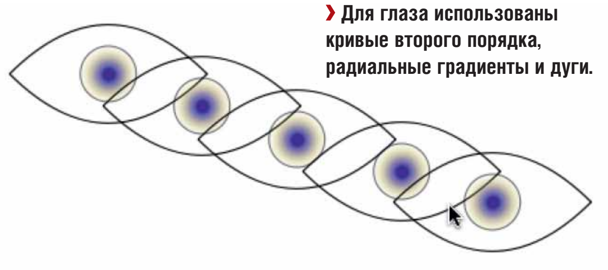
Функции save() и restore() позволяют восстанавливать перенос, поворот, стиль линии и другие настройки, не запоминая каждую из них. Сохраните холст с базовой позицией и стилем, и после рисования окружности, дома, диаграммы или чего угодно вы сможете вызвать restore() и вернуться в исходное

Градиенты, дуги и кривые

Градиенты используются как стили заливки – создайте объект градиента и используйте его для установки fillStyle перед рисованием. Градиенты можно сделать как линейными, так и радиальными, с помощью функций createLinearGradient и createRadilGradient соответственно. Сам градиент – это набор цветовых переходов, добавляемых с помощью addColorStop. Увидеть – значит понять; вот и посмотрите на градиент от красного до зеленого, затем синего и, наконец, белого.

```
gradbox = new iShape();
gradbox.renderShape = function() {
  var gradient = this.c.createLinearGradient(0, 0, 0, 100);
  // создать градиент – градации от x1,y1 до x2,y2
  gradient.addColorStop(0, 'red');
  gradient.addColorStop(0.25, 'green');
  gradient.addColorStop(0.75, 'blue');
  gradient.addColorStop(1, 'white');
  this.c.fillStyle = gradient;
  this.c.fillRect(0, 0, 100, 100);
}
ic.drawShape(gradbox, {x : 300, y : 300});
```

Функция createLinearGradient задает направление градиента – в нашем случае, вертикальное, от 0,0 до 0,100. Каждая из строк .addColorStop устанавливает цвет в соответствующий точке. Ноль (начало) – это красный, сливающийся с зеленым на четверти пути (0.25), затем синий на трех четвертях и белый в конце (1). Поиграйте с цветами и значениями в addColorStop – например, попытайтесь изменить 0.25 на 0.5, а 0.75 на 0.9.



› **Для глаза использованы кривые второго порядка, радиальные градиенты и дуги.**

Радиальные градиенты работают так же, как и линейные, используя «цветовые шаги», но создаваемый градиент не вертикальный, а проходящий от центра одного круга до границы другого, следующим образом:

```
gradblob = new iShape();
gradblob.renderShape = function() {
  var gradient = this.c.createRadialGradient( 50, 50, 1, 50, 50, 100 );
  gradient.addColorStop(0, 'rgba(0,0,228,1)');
  gradient.addColorStop(1, 'rgba(228,0,0,0.5)');
  this.c.fillStyle = gradient;
  this.c.fillRect(0, 0, 100, 100);
}
ic.drawShape(gradblob, {x : 200, y : 200});
```

Синтаксис примерно тот же, так что можете сразу начинать эксперименты с радиальными градиентами. Их можно даже создавать между двумя не выровненными окружностями – попробуйте установить this.c.createRadialGradient(50, 50, 1, 5, 5, 100).

Теперь займемся кривыми. Тэг canvas предлагает несколько способов создания кривых, одна из которых – дуга, позволяющая рисовать круги на холсте:

```
this.c.arc(50, 50, 20, 0, 2*Math.PI, false);
```

Первые и второй аргументы определяют центр дуги, а третий – радиус. Четвертый аргумент – угол начала дуги, пятый – конца. Шестой определяет, рисовать круг по часовой стрелке или против.

Чудненько. Но для рисования более сложных кривых лучше использовать quadraticCurveTo или bezierCurveTo. В следующем примере мы возьмем quadraticCurveTo и нарисуем глаз, точнее, верхнее и нижнее веки. Потом, чтобы правильно выровнять глаза, повернем холст на 45° с помощью функции rotate().

Аргументами для quadraticCurveTo являются координаты x и y исходной точки и точки назначения. Точка назначения определяет, куда вести кривую. Если вы незнакомы с таким типом кривых, посмотрите, как меняется кривая при изменении первых двух аргументов. Добавив градиент к дугам, нарисуем пару-тройку вытаращенных глаз:

```
eye = new iShape();
eye.renderShape = function() {
  this.c.save();
  this.c.beginPath();
  this.c.translate(0, 35);
  this.c.rotate(-45 * Math.PI / 180);
  this.c.moveTo(0, 0);
  this.c.quadraticCurveTo(100, 0, 100, 100);
  this.c.quadraticCurveTo(0, 100, 0, 0);
  this.c.stroke();
  this.c.closePath();
  this.c.beginPath();
  // рисуем дугу
  this.c.arc(50, 50, 20, 0, 2*Math.PI, false);
  this.c.globalAlpha = 0.5;
  var g = this.c.createRadialGradient( 50, 50, 1, 50, 50, 20 );
  g.addColorStop(0, 'rgba(0,0,228,1)');
  g.addColorStop(1, 'rgba(228,199,0,0)');
  //this.c.fillStyle = 'green';
```

Ресурсы

» **http://developer.mozilla.org/en/docs/Canvas_tutorial**
Руководство от Mozilla, описывающее все доступные функции.
» **http://canvaspaint.org**
Полюбуйтесь на canvas-версию старой программы Paint.
» **www.whatwg.org/specs/web-apps/current-work/#the-canvas**
WHATWG-спецификация canvas.

```
  this.c.fillStyle = g;
  this.c.fill();
  this.c.stroke();
  this.c.closePath();
  this.c.restore();
}
ic.drawShape(eye, {x : 100, y : 100});
ic.drawShape(eye, {x : 300, y : 100});
```

Вы, наверное, заметили, что мы ввели новые функции для глаза: beginPath() и closePath(). Path, контур – это то, что функции fill() и stroke() заполняют и вдоль чего рисуют. На глазе мы закрыли один путь и начали другой, так как круги посреди глаза не соединяются с кривыми второго порядка, образующими верхнюю и нижнюю часть глаза. Закомментируйте this.c.closePath() и this.c.beginPath() посреди глаза и посмотрите на результат: у вас получится линия от левой стороны глаза до центра окружности. Так как путь между концом кривой и началом дуги не закрыт, метод stroke соединяет их.

Изображения можно помещать на холст с помощью drawImage(). Эта функция принимает в качестве аргументов объект изображения JavaScript и x / y координаты, но если вы хотите избежать возникновения исключения «не доступно», изображение должно быть загружено до drawImage(). Далее, как и с линиями и заполнением, вы можете повернуть холст до того, как нарисуете изображение, используя функцию .rotate():

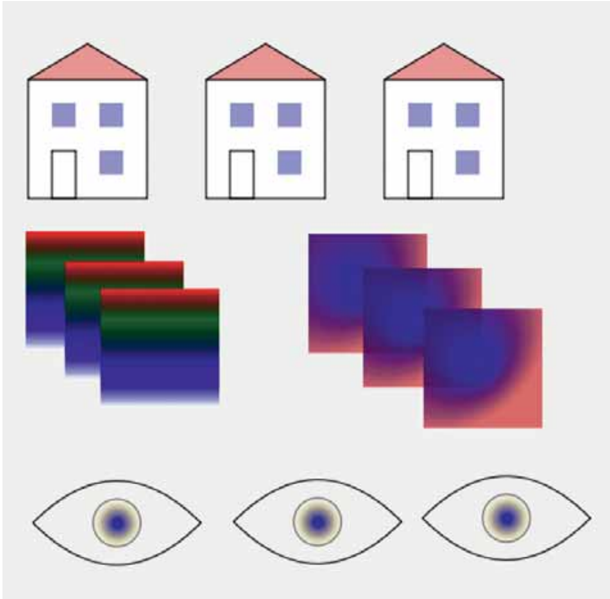
```
this.c.rotate(-45 * Math.PI / 180);
```

Взаимодействие с холстом

Картинки рисовать мы научились. Но ключевая задача сети – обеспечить взаимодействие пользователя с объектами. На нашем уроке мы реализовали возможность создания копий фигур всего одной строкой кода, drawShape(). Форма каждого объекта может быть сколь угодно сложной – изображение, кривые, штрихи и так далее. Чтобы превратить этот скромный базис в простую программу рисования, надо просто следить за событиями мыши. Я собираюсь использовать *Script.aculo.us*, коллекцию интерфейсных библиотек JavaScript, для отслеживания событий мыши, а код поставляется вместе с примером на сайте. Вы можете использовать любую подобную среду.

Цель этого примера – рисовать фигуру на холсте по щелчку мыши с помощью drawShape(). На текущий момент единственной фигурой является домик. Мы будем слушать любые щелчки мыши для id canvas.

```
function setupInteraction() {
  currentDrawShape = house;
  new Event.observe(
    'canvas',
    'click',
    function(evt) {
      ic.drawShape(currentDrawShape, {x : Event.pointerX(evt), y :
      Event.pointerY(evt)});
    }
  );
}
```



› **Проект завершен: несколькими строками кода вы можете создавать простые приложения для рисования любых изображений и диаграмм.**

Вызов setupInteraction() в конце функции draw() позволяет помещать объекты на холст с помощью мыши. Пока от него мало проку, разве что вы затеяли нарисовать много-много домиков. Куда полезнее будет позволить пользователю самому выбирать из доступных фигур-шаблонов – как в *Dia* или *OmniGraffe*. На уровне кода, для того, чтобы изменить изображение, которое мы рисуем, достаточно изменить значение currentDrawImage. Поэтому, если вы добавите строку

```
currentDrawShape = eye;
```

в setupInteraction(), то будете рисовать глаза вместо домов.

Включим функцию addOption() для добавления фигуры в перечень шаблонов:

```
function addShapeOption(shape, name) {
shapeOptions[name] = shape;
$('slist').innerHTML += '<li><a href="#" onClick="currentDrawSh
ape = shapeOptions[\''+name+'\']; return false;">'+name+'</li>';
}
```

Это хороший способ для пользователя выбрать текущий шаблон currentDrawShape, и все, что нам надо сделать для активации опции, это вызвать функцию

```
addShapeOption(theShapeObject, 'name of object');
```

В заключительной версии исходного кода на сайте каждая фигура сама добавляет себя в список, а пользователь может из него выбирать.

Тэг canvas будет одним из самых интересных добавлений в спецификацию HTML, с учетом огромного интереса к нему в так называемых средах Web 2.0. Ныне доступные примеры покамест довольно примитивны, но потенциал огромен. Уже создаются библиотеки для графиков, и комбинируя их с простыми действиями, можно добиться поразительных – и полезных – приложений. LXF

Попробуйте еще

Не останавливайтесь на достигнутом – попробуйте другие функции...
» createPattern() превращает изображения или другие объекты canvas в шаблоны для многократного использования.
» clearRect(x2,y1,x2,y2) очищает кусок холста.
» clip() создает путь отсечения так, что вы можете ограничить холст определенной фигурой, например, кругом или сложным многоугольником.

» **Через месяц** Сложные скрипты DCOP* (*если они не заработают, придумаем что-нибудь другое)

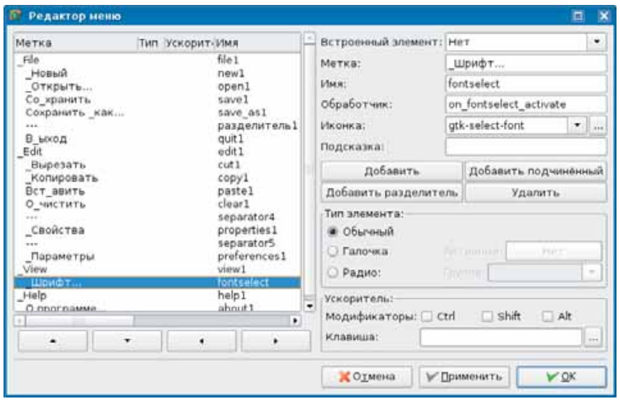
Погружение

ЧАСТЬ 8 Мы уже успели замочить ножки в океане возможностей, которые предоставляет GNOME – и сейчас самое время зайти поглубже. **Андрей Боровский** проверяет, теплая ли водичка.

В прошлый раз мы начали работу над демо-приложением GNOME – текстовым редактором. До сих пор в этой программе собственно «гномовского» было не так уж много: только код инициализации. На этот раз мы рассмотрим некоторые функции, относящиеся исключительно к GNOME; но сначала – немного визуального программирования.

Редактирование меню

Заготовка главного меню нашей программы содержит пункт Вид (View), однако само это меню пустое. Добавим в меню Вид новую команду Шрифт..., позволяющую выбрать шрифт для отображения текста. Щелкните правой кнопкой мыши по строке главного меню в окне формы приложения и в открывшемся контекстном меню выберите команду Правка меню... (**Рис. 1**).



› Рис. 1. Редактор меню *Glade*.

Список названий пунктов меню отражает существующие между ними иерархические отношения. Сдвиг строки вправо означает переход к следующему уровню вложенности меню. С помощью кнопок Добавить и Добавить подчиненный в меню можно добавить новый пункт. При добавлении нового подчиненного пункта необходимо указать для него родительский пункт меню. Если мы потом передумаем, то с помощью клавиш навигации сможем переместить уже созданный пункт в другую группу или на другой уровень вложенности.

Раскрывающийся список Встроенный элемент позволяет присвоить новому пункту меню свойства одного из стандартных элементов меню (эти встроенные элементы используют пункты меню, созданные автоматически). Строка ввода Метка позволяет указать название пункта меню. Нижний дефис в названии пункта меню отмечает подчеркнутый символ, который нужно вводить в сочетании с Alt для быстрого доступа к этому пункту. Строка ввода имя позволяет указать имя объекта GtkImageMenuItem, соответствующего пункту меню. В строке ввода Обработчик указывается имя функции-обработчика сигнала activate, посылаемого командой меню. Комбинированный раскрывающийся список Иконка позволяет указать пиктограмму, которая будет отображаться рядом с командой меню. Можно загрузить собственную пиктограмму из файла либо использовать одну из стандартных пиктограмм, установленных в системе.

Для создаваемого нами пункта меню Шрифт... выберем стандартную пиктограмму *gtk-select-font*. Остальные элементы редактора меню нам сейчас не интересны, так что мы их опустим (читателю, как всегда, рекомендуется обратиться к документации). Щелкните кнопку OK. Теперь у нас есть новая команда меню, для которой нужно написать обработчик (функция on_fontselect_activate()). Для выбора шрифта используйте уже знакомое нам диалоговое окно GtkFontSelectionDialog. Полный код обработчика вы найдете на диске, мы на нем останавливаться не будем.

Буфер обмена GTK+ и GNOME

Интерфейс буфера обмена в *GTK+* основан на структуре GtkClipboard. Мы получаем доступ к этой структуре с помощью функции gtk_clipboard_get(). Первый вызов gtk_clipboard_get() создает экземпляр структуры, которая переходит под управление *GTK+* (так что мы не должны удалять ее явным образом). При последующих вызовах gtk_clipboard_get() возвращает указатель на существующую структуру GtkClipboard.

При работе с X Window следует учесть, что система поддерживает как минимум два буфера обмена (теоретически их может быть и больше). Если вы используете команду меню Правка | Копировать, данные обычно попадают в буфер обмена, связанный с атомом CLIPBOARD. В то же время любые данные, выделенные мышью в окне X-программы, заносятся в буфер обмена, связанный с атомом PRIMARY (вставка данных из этого буфера обычно выполняется с помощью щелчка средней кнопкой мыши). Вообще говоря, X-программы вольны интерпретировать оба буфера так, как им заблагорассудится. На практике это иногда приводит к тому, что результаты вставки данных из буфера обмена оказываются несколько неожиданными.

Месяц назад Мы изучали среду *Anjuta* и невзначай написали простой текстовый редактор.

LINUX FORMAT на DVD
» Код примеров статьи

в GNOME

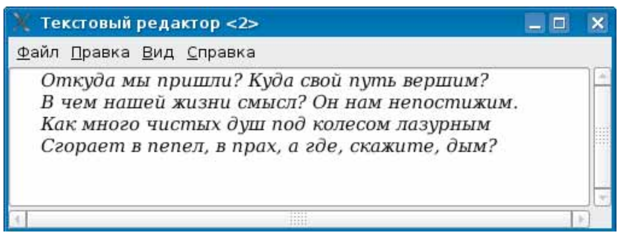
Интерфейс буфера обмена *GTK+* позволяет работать со всеми буферами обмена *X*. *GTK+* поддерживает размещение данных в буферах обмена в нескольких форматах и отложенную запись.

Организовать передачу данных между буфером обмена и экземпляром GtkTextBuffer очень просто. Ниже приводится текст обработчика сигнала меню copy1_activate нашего текстового редактора (этот сигнал генерирует команда Правка | Копировать).

```
void on_copy1_activate (GtkMenuItem * menuitem, gpointer user_data)
{
char * atom = gdk_atom_name("CLIPBOARD");
GtkClipboard * cb = gtk_clipboard_get(atom);
g_free(atom);
gtk_text_buffer_copy_clipboard(textbuffer, cb);
}
```

Вызывая gtk_clipboard_get(), мы передаем этой функции X-атом, идентифицирующий буфер обмена, который мы хотим открыть, и получаем строку с именем атома с помощью функции gdk_atom_name(). Функция gtk_text_buffer_copy_clipboard() копирует в буфер обмена строку текста, выделенную в текстовом буфере GtkTextBuffer (программно или с помощью интерфейса пользователя). Первым параметром функции, копирующей данные, должен быть указатель на объект GtkTextBuffer, вторым параметром – указатель на объект GtkClipboard. Для вставки данных из буфера обмена в текстовый буфер применяется функция gtk_text_buffer_paste_clipboard(), которой, помимо прочего, следует передать итератор, указывающий, где именно в тексте должна быть вставлена строка из буфера обмена (при использовании текстового буфера вместе с GtkTextView вместо итератора функции можно передать NULL).

Разумеется, у буфера обмена есть и собственный API, который можно применять независимо от других компонентов *GTK+*/GNOME. Для передачи в буфер обмена строки текста в общем случае используется функция gtk_clipboard_set_text(). Первым параметром этой функции должен быть указатель на объект GtkClipboard. Второй и третий параметры, соответственно, строка текста с нулевым конечным символом и длина строки в байтах (напомню, что в *GTK+* и GNOME по умолчанию используется кодировка UTF-8). Для передачи в буфер обмена растрового графического объекта служит функция gtk_clipboard_set_image(). У этой функции всего два параметра: указатель на объект GtkClipboard и указатель на объект GdkPixBuf, содержащий данные растрового изображения.



› **Рис. 2.**

Если вы хотите поместить в буфер обмена данные в нескольких форматах или реализовать отложенную запись, все становится более сложным. В этом случае ваша программа должна объявить системе, что она готова передать в буфер обмена данные в определенных форматах и зарегистрировать функции обратного вызова, которые будут выполнять их фактическую обработку. Когда данные понадобятся, эти функции будут вызваны системой. Рассмотрим пример – фрагмент программы, копирующей в буфер строку текста:

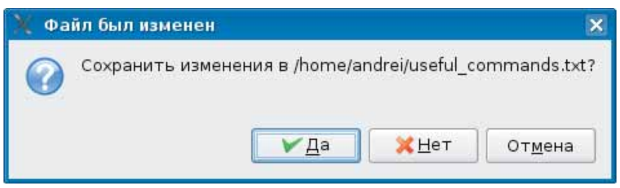
```
void cb_get_func(GtkClipboard * clipboard, GtkSelectionData * selection_data,
guint info, gpointer user_data)
{
char * str = "Данные для буфера обмена";
gtk_selection_data_set_text(selection_data, str, strlen(str));
}

void cb_clear_func(GtkClipboard * clipboard, gpointer user_data)
{
// Nothing to do
}

void on_copy1_activate (GtkMenuItem * menuitem, gpointer user_data)
{
static const GtkTargetEntry targets[] =
 { { "UTF8_STRING", 0, GDK_TARGET_STRING } };
char * atom = gdk_atom_name("CLIPBOARD");
GtkClipboard * cb = gtk_clipboard_get(atom);
g_free(atom);
gtk_clipboard_set_with_data(cb, targets, 1, cb_get_func, cb_clear_func, NULL);
}
```

Процесс записи данных в буфер обмена инициирует все тот же обработчик on_copy1_activate(). Регистрация форматов данных в буфере обмена начинается с создания списка форматов в виде массива структур GtkTargetEntry. Мы немного упростили себе жизнь за счет того, что сконструировали этот массив (который в нашем случае состоит из одного элемента) статически. В сложных программах список форматов, в которых копируются данные, может быть заранее неизвестен, и тогда его придется создавать динамически, с помощью интерфейса структуры GtkTargetList.

Получив указатель на объект GtkClipboard, мы вызываем функцию gtk_clipboard_set_with_data(). Эта функция регистрирует список фор- »



› **Рис. 3.**

» матов данных, которые программа может передать в буфер обмена (третий параметр функции – количество элементов в массиве targets), и передает системе адреса двух функций обратного вызова. Первая функция (cb_get_func()) будет вызвана для того, чтобы передать в буфер обмена данные в запрошенном формате. Вторая функция (cb_clear_func()) должна оповестить программу о том, что содержимое буфера обмена было перезаписано другим приложением (и ресурсы, выделенные нашей программой для буфера обмена, можно освободить). Последний параметр функции gtk_clipboard_set_with_data() – указатель, который будет передан функциям обратного вызова: его можно использовать для передачи функциям произвольных данных. Поскольку в нашем примере мы не выделяли динамически никаких ресурсов для работы с буфером, в функции cb_clear_func() не нужно ничего и высвобождать.

Рассмотрим подробнее функцию cb_get_func(). Помимо прочего, этой функции передается указатель на объект GtkSelectionData. Данные записываются в буфер с помощью функций, связанных с этим объектом. Для записи данных в текстовом формате используется функция gtk_selection_data_set_text(). Запись в буфер обмена растровой картинки выполняется функцией gtk_selection_data_set_pixbuf(), а для передачи списка ссылок на ресурсы (сетевые или файловые) применяется функция gtk_selection_data_set_uris(). Обратите внимание на параметр info функции cb_get_func(). Этот параметр содержит идентификатор формата, в котором система желает получить данные. Поскольку мы зарегистрировали только один формат, значение info можно не проверять. В параметре user_data функциям обратного вызова передается указатель на дополнительные данные. Зарегистрировать форматы данных и функции обратного вызова можно также с помощью функции gtk_clipboard_set_with_owner(), которая отличается от gtk_clipboard_set_with_data() только тем, что в качестве дополнительного параметра функциям обратного вызова передается указатель GObject *, а не gpointer.

Для вставки данных из буфера обмена также реализованы два интерфейса. Вы можете зарегистрировать функцию, которая будет вызвана системой при появлении в буфере обмена данных. Регистрация функций обратного вызова для вставки данных выполняется с помощью функций семейства gtk_clipboard_request_XXX(). Хотя функция обратного вызова регистрируется gtk_clipboard_request_XXX() для приема определенного типа данных, она будет вызвана в ответ на передачу в буфер любых данных. Если в буфере обмена появились данные в неподходящем формате, функции обратного вызова будет передано значение **NULL**. Если вы не хотите иметь дело с обратными вызовами, воспользуйтесь функциями gtk_clipboard_wait_for_text() и gtk_clipboard_wait_for_image() и им подобными. Эти функции сами возвращают данные из буфера обмена и не требуют использования обратных вызовов.

Виртуальная файловая система GNOME

Как и KDE, GNOME стремится объединить файловые ресурсы сети и локальные файловые ресурсы в единое файловое пространство. С этой целью в GNOME, как и в KDE, реализован дополнительный уровень абстракции поверх файловой системы. Виртуальная файловая система GNOME (GNOME VFS) предоставляет единый интерфейс программирования для доступа к самым разным ресурсам, которые могут быть представлены в виде файлов, а также включает множество функций, упрощающих работу с файлами.

Рассмотрим функцию open_file() нашего текстового редактора (листинг сокращен):

```
void open_file(char * file_name)
{
GnomeVFSHandle * file_handle;
GnomeVFSFileInfo * file_info;
GnomeVFSFileSize bytes_read;
gpointer buf;
if (gnome_vfs_open(&file_handle, file_name, GNOME_VFS_OPEN_READ)
!=
 GNOME_VFS_OK) {
// Сообщить пользователю об ошибке.
}
strncpy(filename, file_name, 256);
gtk_text_buffer_set_text(textbuffer, "", 0);
file_info = gnome_vfs_file_info_new();
gnome_vfs_get_file_info_from_handle(file_handle, file_info,
GNOME_VFS_FILE_INFO_DEFAULT);
buf = g_malloc(file_info.size);
gnome_vfs_read(file_handle, buf, file_info.size, &bytes_read);
gtk_text_buffer_set_text(textbuffer, buf, bytes_read);
g_free(buf);
gnome_vfs_file_info_unref(file_info);
gnome_vfs_close(file_handle);
gtk_text_buffer_set_modified(textbuffer, FALSE);
}
```

Функции GNOME VFS API объявлены в заголовочном файле **libgnomevfs/gnome-vfs.h** (сама библиотека *libgnomevfs* включена в проект приложения GNOME, созданный в *Anjuta*, по умолчанию). Систему GNOME VFS следует инициализировать при помощи функции gnome_vfs_init() (эту функцию достаточно вызвать один раз). По окончании работы с GNOME VFS мы вызываем функцию gnome_vfs_shutdown(). Для работы с файлами GNOME VFS предоставляет набор функций, похожих на функции POSIX.

Файл открывается с помощью функции gnome_vfs_open(). В отличие от функций POSIX, gnome_vfs_open() не возвращает дескриптор файла: функция возвращает константу, указывающую статус завершения операции. Роль дескриптора файла при работе с функциями GNOME VFS выполняет значение типа VFSHandle (адрес переменной, в которую будет записан указатель на переменную типа VFSHandle, передается функции gnome_vfs_open() в первом параметре). Второй параметр gnome_vfs_open() – имя открываемого файла, а в третьем параметре функции передаются флаги, описывающие то, для чего открывается файл (GNOME_VFS_OPEN_READ – для чтения, GNOME_VFS_OPEN_WRITE – для записи, и т.д.).

Вы можете преобразовать дескриптор файла в VFSHandle с помощью функции gnome_vfs_open_fd(). Функция gnome_vfs_read() предназначена для чтения данных из файла. Как и gnome_vfs_open(), она возвращает информацию о статусе выполняемой операции. Все остальные результаты передаются через параметры. Первый параметр gnome_vfs_read() – указатель на VFSHandle файла, из которого нужно читать данные. Далее следуют указатель на буфер для данных, размер буфера и указатель на переменную, в которую функция запишет, сколько байтов ей удалось прочитать. Для записи данных используется функция gnome_vfs_write() с тем же списком параметров, что и у gnome_vfs_read().

Закончив работу с файлом, мы закрываем его, используя функцию gnome_vfs_close(). Получить информацию об открытом файле (по крайней мере, о файле локальной файловой системы) можно с помощью функции gnome_vfs_get_file_info_from_handle(). Ее первым параметром должен быть указатель на идентификатор открытого файла VFSHandle, вторым параметром – указатель на экземпляр структуры GnomeVFSFileInfo, в который будут записаны сведения о файле, а третьим параметром – комбинация флагов, указывающих, какие именно данные мы хотим получить.

Экземпляр GnomeVFSFileInfo создается с помощью функции gnome_vfs_file_info_new(). Функция gnome_vfs_file_info_unref() уменьшает внутренний счетчик ссылок структуры GnomeVFSFileInfo и удаляет структуру при обнулении счетчика. Структура GnomeVFSFileInfo содержит множество полей, но не все они заполняются по умолчанию (это сделано, в том числе, ради быстродействия). Специальные флаги заставляют функцию gnome_vfs_get_file_info_from_handle() возвращать дополнительные сведения о файле. Помимо функции gnome_vfs_get_file_info_from_handle(), есть еще функция gnome_vfs_get_file_info(), позволяющая получить информацию о файле, заданным своим именем. Для работы со структурой GnomeVFSFileInfo определено несколь-

ко дополнительных функций, более или менее подробное описание которых можно найти в документации GNOME.

Чтобы создать новый или перезаписать уже существующий файл, применяется функция gnome_vfs_create(). Первые три параметра у этой функции те же, что и у gnome_vfs_open(). В четвертом параметре функции передается значение типа gboolean, указывающее, должна ли функция перезаписать файл, если он уже существует. В пятом параметре следует передать маску разрешений для создаваемого файла. У функций GNOME VFS, работающих с именем файла в текстовом формате, есть аналоги с окончанием _uri. Этим функциям вместо строки с именем файла передается указатель на структуру GnomeVFSURI, которая упрощает работу с универсальными идентификаторами ресурсов.

В чем же преимущество функций GNOME VFS API перед функциями POSIX? Предположим, нам нужно загрузить в с Web-сайта документ www.somesite.com/somefile.txt. Нет ничего проще – открываем файл для чтения:

```
gnome_vfs_open(&file_handle, "http://www.somesite.com/somefile.txt",
GNOME_VFS_OPEN_READ);
```

Префикс **http://** надо указывать обязательно, иначе система не узнает, какой модуль виртуальной файловой системы должен использоваться для загрузки ресурса. Дальше данные можно считывать с помощью gnome_vfs_read(). Загрузить файл на FTP-сайт тоже очень просто. Создаем или перезаписываем файл, например:

```
gnome_vfs_create(&file_handle, "ftp://user:password@somesite.com/
home/user/somefile.txt", GNOME_VFS_OPEN_WRITE, FALSE, 0666);
```

Затем данные записываются в новый файл с помощью gnome_vfs_write() (для работы с идентификаторами FTP-ресурсов лучше использовать структуру GnomeVFSURI и соответствующие функции). Благодаря GNOME VFS мы получаем возможность использовать единый интерфейс и одну программную логику для доступа к самым разным ресурсам. Отличия могут быть связаны только с быстродействием. Если программа ориентирована прежде всего на работу с локальными файлами, чаще используют блокирующие функции в главном потоке (как и мы в рассмотренном примере). Если программа нацелена на работу с ресурсами Глобальной сети, лучше использовать потоки GNOME или асинхронные функции ввода/вывода GNOME VFS.

Единый интерфейс работы с разными ресурсами реализуют модули GNOME VFS. Модули GNOME VFS поддерживают работу с HTTP, FTP, защиту данных с помощью SSL и низкоуровневую работу с сокетами. Вообще говоря, модули GNOME VFS позволяют представить в виде файловых систем множество самых разных источников данных, а если нужного именно вам модуля в системе нет, вы можете написать его сами – это не так уж и трудно. GNOME VFS позволяет конвейеризовать консольные утилиты прямо в строке идентификатора ресурса с помощью разделителя #. Таким образом, можно приказать системе, чтобы она загрузила с web-сайта архив **tar.gz**, распаковала его на лету и передала программе данные для чтения.

С помощью GNOME VFS можно работать с MIME-типами, осуществлять мониторинг изменения файлов, директорий и монтирования файловых систем и многое-многое другое.

Есть у GNOME VFS и набор специальных функций для работы с директориями. Рассмотрим фрагмент консольной программы (графический интерфейс? Зачем нам графический интерфейс?) *scandir*. Функция scan_directory(), определенная в этой программе, сканирует директорию, имя которой передано ей в ее единственном параметре, и распечатывает данные о содержащихся в этой директории файлах. Рассмотрим сокращенный листинг этой функции (полный вариант, как всегда, на диске).

```
int scan_directory(char * dir_name)
{
GnomeVFSDirectoryHandle * handle;
GnomeVFSFileInfo * file_info;
GnomeVFSResult res;
if ((res = gnome_vfs_directory_open(&handle, dir_name,  GNOME_VFS_
FILE_INFO_FOLLOW_LINKS)) != GNOME_VFS_OK) {
g_print("failed to open the directory: %s\n",
gnome_vfs_result_to_string(res));
return res;
}
file_info = gnome_vfs_file_info_new();
while ((res = gnome_vfs_directory_read_next(handle, file_info)) ==
GNOME_VFS_OK) {
g_print("name: %s\n", file_info->name);
g_print("type: ");
switch(file_info->type) {
case GNOME_VFS_FILE_TYPE_REGULAR:
g_print("file\n");
break;
...
case GNOME_VFS_FILE_TYPE_UNKNOWN:
g_print("unknown\n");
break;
}
g_print("size: %lli\n\n", file_info->size);
gnome_vfs_file_info_clear(file_info);
}
gnome_vfs_directory_close(handle);
gnome_vfs_file_info_unref(file_info);
if (res != GNOME_VFS_ERROR_EOF) {
g_print("Error reading the directory: %s\n",
gnome_vfs_result_to_string(res));
return res;
}
   return 0;
}
```

Мы открываем директорию для чтения с помощью функции gnome_vfs_directory_open(). Открытая директория идентифицируется в GNOME VFS значением типа GnomeVFSDirectoryHandle (адрес переменной, которой должен быть присвоен указатель на это значение, передается как первый параметр функции gnome_vfs_directory_open()). Во втором параметре функции передается имя директории, которую вы хотите открыть, а в третьем – набор флагов, определяющих, какие данные о дочерних элементах директории вы хотите получать (эти флаги – того же типа, что и флаги функции vfs_get_file_info()).

Как и все функции GNOME VFS API, gnome_vfs_directory_open() возвращает численный код, свидетельствующий о том, была ли она выполнена и какие возникли ошибки. Мы можем получить соответствующую этому коду строку пояснения на английском языке с помощью функции gnome_vfs_result_to_string(). Когда работа с директорией закончена, она закрывается с помощью функции gnome_vfs_directory_close(). Для чтения содержимого директории предназначена функция gnome_vfs_directory_read_next(), которая записывает в переменную file_info информацию о следующем элементе каталога и возвращает значение GNOME_VFS_ERROR_EOF, если прочитана вся директория. Мы считываем информацию о директории из полей file_info. Если одна и та же структура GnomeVFSFileInfo используется в программе несколько раз, перед повторным использованием ее следует очищать с помощью функции gnome_vfs_file_info_clear().

Конечно, в программах GNOME можно обойтись и без GNOME VFS, но использование этой системы позволяет значительно расширить функциональность программы без особых трудозатрат со стороны программиста. Библиотеки GNOME VFS – лишь часть обширного набора библиотек GNOME. Вспомогательным библиотекам GNOME будет посвящена следующая, заключительная статья этой серии. LXF

Через месяц Мы разберемся с наиболее интересными библиотеками GNOME и подключим к работе C++.

Команды и фабрики

LINUX FORMAT на DVD
» Код примеров статьи

ЧАСТЬ 6 Антон Черноусов готов познакомить вас с очередной партией паттернов, которые помогут сделать ваши приложения еще более гибкими и расширяемыми.

Вместо предисловия

В предыдущей статье мой коллега Александр Бабаев рассмотрел вопросы организации и использования БД в Java-приложениях, и в том числе вопросы подключения к БД посредством ConnectionPool.

Сегодня мы рассмотрим применение двух паттернов, безусловно, оказавших огромное воздействие на проектирование систем – Command и Factory Method. Их применение позволит сделать ваше приложение расширяемым.

Команды

В LXF92 мы кратко описали стратегии, предназначенные для реализации «Контроллера», и обещали более подробно рассмотреть стратегию Command and Controller. Чтобы выполнить это обещание, нам придется сначала познакомиться с паттерном Command.

Задача, которая стоит перед контроллером (сервлетом) при получении управляющего сигнала, как правило, заключается в выполнении последовательности действий, часто атомарной (т.е. обрабатываемой как единое целое). Например, в сервлете AddressBookServlet, реализованном в предыдущей статье, метод handleEdit вызывается, когда адрес на который обращается пользователь – это "/edit".

К сожалению, на примере AddressBookServlet мы видим, что при увеличении функциональности web-приложения растёт и количество методов, реализованных в сервлете; класс «засоряется», код становится менее структурированным и читабельным. Решить проблемы с кодом можно, «обернув» методы в специальные классы, которые будут выполнять атомарные операции и предоставлять сервлету стандартный интерфейс, предназначенный для этих целей. Для выполнения поставленной задачи воспользуемся паттерном Command.

Команды удобны прежде всего тем, что они маскируют конкретную реализацию, находящуюся за интерфейсной прослойкой. Интерфейс остается одним и тем же, независимо от того, с чем работает команда [1].

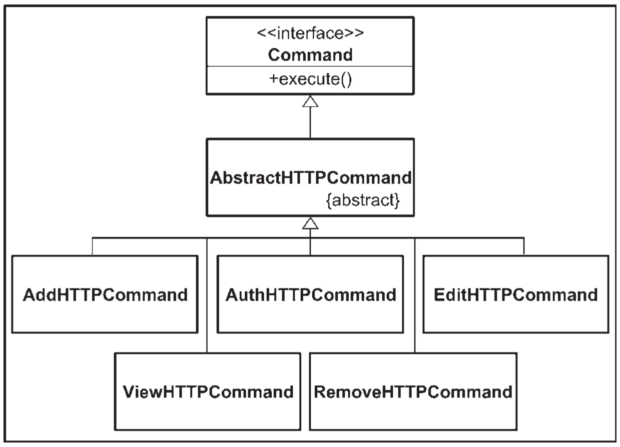
```
public interface Command {
    public void execute() throws Exception;
}
```

Выше представлен простой интерфейс Command всего с одним методом execute(), который и олицетворяет идею одноименного паттерна. Он будет нашим стандартым интерфейсом. Более сложная реализация интерфейса включает метод unexecute(). Класс, реализующий интерфейс Command, инкапсулирует в методе execute() обозначенные выше атомарные операции, а в методе unexecute() реализуется механизм отмены. Часто методы execute() и unexecute() называют do() и undo(), соответственно.

Выделим команды, которые нам необходимо реализовать (отметим удачное разделение на методы): Add, Auth, Edit, View, Remove. Учитывая то, что у наших команд будут некоторые идентичные методы и атрибуты, предлагаю создать абстрактный класс AbstractHTTPCommand, в котором они будут собраны. Класс будет реализовывать интерфейс Command, и его наследование автоматически позволит обеспечить необходимый уровень интеграции. Итак, в основе каждой команды

» **Месяц назад** Мы освоили подключение к базам данных.

будет лежать абстрактный класс AbstractHTTPCommand, в котором реализован интерфейс для выполнения операций (на **Рис. 1** вы можете видеть диаграмму классов команд нашего приложения).



› Рис. 1. Диаграмма классов.

Определим общие методы для абстрактного класса: initCommand() – предназначен для инициализации команды, makeDataToView() – для подготовки данных для отображения в случае их изменения, outputPage() – метод для переадресации пользователя (он будет перенесен из AddressBookServlet без изменений) и другие. Ниже представлена реализация методов initCommand() и makeDataToView():

```
protected void initCommand(ServletContext sc, HttpServletRequest
aRequest,
        HttpServletResponse aResponse, String viewPath,
        String resultPath, String errorPath) {
    this.setSc(sc);
    this.setARequest(aRequest);
    this.setAResponse(aResponse);
    this.setResultPath(resultPath);
    this.setErrorPath(errorPath);
    this.setViewPath(viewPath);
}

public void makeDataToView() {
    Map<String, String> numbers = new HashMap<String, String>();
    Map<String, String> comments = new HashMap<String, String>();

    for (Map.Entry<String, Contact> entry :
            _addressBook.getContacts().entrySet()) {
        numbers.put(entry.getKey(), entry.getValue().getNumber());
        comments.put(entry.getKey(), entry.getValue().getComment());
    }

    aRequest.setAttribute("numbers", numbers);
    aRequest.setAttribute("comments", comments);

    if (aRequest.getAttribute("message") == null) {
        aRequest.setAttribute("message", "");
    }
}
```

Метод makeDataToView() – существенная часть метода handleView() класса AddressBookServlet. Вы можете удивиться, для чего метод initCommand() содержит так много параметров; это необходимо для того, чтобы в момент создания команды полностью передать ей всю необходимую для ее выполнения информацию. Параметры *viewPath*, *resultPath* и *errorPath* появились не случайно – они предназначены для адресов (видов, если использовать термины MVC), используемых в случае простого отображения данных, удачного и, соответственно, неудачного выполнения команды.

Перейдем к реализации самих команд. Рассмотрим, например, метод execute() класса EditHTTPCommand. Он практически полностью соответствует первоначальному методу handleEdit класса AddressBookServlet, исключая переадресацию пользователя на конкретный вид.

```
public void execute() throws Exception {
    if (aRequest.getParameter("number") == null) {
        _addressBook.removeContactByNumber(aRequest.getParameter("
number"));
        aRequest.setAttribute("message",
                "Не определено, что редактировать");
        outputPage(this.getErrorPath(), aRequest, aResponse);
    } else if (aRequest.getParameter("edited") != null) {
        _addressBook.editContactByNumber(aRequest.
getParameter("edited"),
            aRequest.getParameter("name"),
            aRequest.getParameter("number"),
            aRequest.getParameter("comment"));
        aRequest.setAttribute("message", "Контакт \"" +
            aRequest.getParameter("name") + "\" отредактирован");
        makeDataToView();
        outputPage(this.getResultPath(), aRequest, aResponse);
    } else {
        Contact contact =
         _addressBook.getContactByNumber(aRequest.getParameter("nu
mber"));
        aRequest.setAttribute("action", "edit");
        aRequest.setAttribute("edit.name", contact.getName());
        aRequest.setAttribute("edit.number", contact.getNumber());
        aRequest.setAttribute("edit.comment", contact.getComment());
        outputPage(this.getViewPath(), aRequest, aResponse);
    }
}
```

Остальные команды реализуются аналогичным образом. Исключение из общего процесса рефакторинга кода составит команда ViewHTTPCommand, для которой уже реализована большая часть функционала:

```
public void execute() throws Exception {
    makeDataToView();
    outputPage(this.getViewPath(), aRequest, aResponse);}
```

Выполнение любой команды унифицировано и будет выглядеть следующим образом:

```
    Command cmd;
    cmd = new EditHTTPCommand(this.getServletContext(),
            aRequest, aResponse, "edit.jsp", "view.jsp", "view.jsp");
    try {
        cmd.execute();
    } catch (Exception e) {
        e.printStackTrace();
}
```

После создания всех команд контроллер должен преобразиться: лишние методы уйдут, а его главная функция – управление – будет восстановлена:

```
private void handle(HttpServletRequest aRequest,
        HttpServletResponse aResponse) throws ServletException,
IOException {
        aRequest.setCharacterEncoding("utf-8");
        String target =
        aRequest.getRequestURI().substring(aRequest.getContextPath().
length());
        Command cmd;
        if (target.equals("/")) {
            cmd = new ViewHTTPCommand(this.getServletContext(),
                aRequest, aResponse, "index.jsp", null, null);
        } else if ("/add".equals(target)) {
```

```
            cmd = new AddHTTPCommand(this.getServletContext(),
                      aRequest, aResponse, "edit.jsp", "view.jsp", null);
        } else if ("/view".equals(target)) {
            cmd = new ViewHTTPCommand(this.getServletContext(),
                      aRequest, aResponse, "view.jsp", null, null);
        } else if ("/edit".equals(target)) {
            cmd = new EditHTTPCommand(this.getServletContext(),
                      aRequest, aResponse, "edit.jsp", "view.jsp", "view.jsp");
        } else if ("/remove".equals(target)) {
            cmd = new RemoveHTTPCommand(this.getServletContext(),
                      aRequest, aResponse, null, "view.jsp", null);
        } else if ("/auth".equals(target)) {
            cmd = new AuthHTTPCommand(this.getServletContext(),
                     aRequest, aResponse, "auth.jsp", "index.jsp", "auth.jsp");
        } else {
            cmd = new ViewHTTPCommand(this.getServletContext(),
                      aRequest, aResponse, "view.jsp", null, null);
        }
        try {
            cmd.execute();
        } catch (Exception e) {
            // oopst...
            e.printStackTrace();
        }
    }
```

С одной стороны, от внедрения паттерна Command в наше web-приложение мы получили следующие преимущества: управление сосредоточилось в одном месте (в методе handle(), представленном выше), код стал более структурированным. С другой стороны, мы пока не добились возможности полного отторжения продукта от разработчика (при котором исходный код не передается заказчику), так как для изменения функциональности или доработки приложения (как минимум, при добавлении новой команды) необходимо производить перекомпиляцию.

В принципе, ничего непреодолимого нет, в любую программу можно внести исправления, но в одни они вносятся проще, чем в другие. Поставим себе задачу сделать внесение исправлений в наше приложение простым. Для этого нам надо:

- Вынести связи между адресами, командами и видами за пределы приложения.
- Заложить возможность создания экземпляра команды при наличии ее названия.
- Изменить процесс вызова конкретной команды, чтобы при расширении приложения (дополнении новых команд) не требовалось вносить изменения в уже существующий код.

Хранение настроек команд

Для решения первой из поставленных задач мы не будем «изобретать велосипед» и воспользуемся средствами конфигурирования web-приложения. В файл **web.xml** введем параметры для нашего контроллера:

```
<servlet>
  <display-name>AddressBook</display-name>
  <servlet-name>ABServlet</servlet-name>
  <servlet-class>AddressBookServlet</servlet-class>
  <init-param>
       <param-name>/</param-name>
       <param-value>root</param-value>
  </init-param>
  <init-param>
       <param-name>rootCommand</param-name>
       <param-value>ViewHTTPCommand</param-value>
  </init-param>
  <init-param>
       <param-name>rootView</param-name>
       <param-value>index.jsp</param-value>
  </init-param>
  <init-param>
       <param-name>rootResult</param-name>
       <param-value>null</param-value>
  </init-param>
  <init-param>
       <param-name>rootError</param-name>
       <param-value>null</param-value>
  </init-param>
  <load-on-startup>0</load-on-startup>
</servlet>
```

Для каждого адреса вводится параметр, который мы будем называть ключом (для «/» это «root»). Ключ с добавлением суффикса (Command, View, Result, Error) обозначает конкретный параметр, значение которого мы будем извлекать прямо в приложении. Для получения необходимых данных можно воспользоваться следующим кодом (хотя мы в дальнейшем будем действовать по-другому):

```
ServletContext sc = this.getServletContext();
    String key = sc.getInitParameter("/");
    String commandName = sc.getInitParameter(key + "Command");
    String commandView = sc.getInitParameter(key + "View");
    String commandResult = sc.getInitParameter(key + "Result");
    String commandError = sc.getInitParameter(key + "Error");
```

Аналогичные настройки необходимо сделать для всех адресов, с которыми работает наше приложение («/», «/add», «/auth», «/edit», «/remove», «/view»). Включение в файл конфигурации этих значений позволит настраивать web-приложение без исправлений в коде.

Создание экземпляра класса по имени

Экземпляр класса можно создавать при наличии полного имени, записанного в строке, с помощью методов forName() и newInstance() класса Class (например, для класса Date: Object o = Class.forName("java.util.Date").newInstance()). Если при создании объекта на основе имени класса конструктору необходимо передать ряд параметров, то воспользоваться данным методом нельзя, но создать объект возможно с помощью класса java.lang.reflect.Constructor, например:

```
Class c = Class.forName(key);
Class[] parameterTypes = new Class[3];
parameterTypes[0] = String.class;
parameterTypes[1] = String.class;
parameterTypes[2] = String.class;
Constructor constructor = c.getConstructor(parameterTypes);
Object[] args = new Object[3];
args[0] = "par1";
args[1] = "par2";
args[2] = "par3";
Object o = constructor.newInstance(args);
```

Давайте разберем приведенный пример. Сначала мы создаем экземпляр класса Class по имени key, далее выделяем массив из типов параметров в том количестве и последовательности, как они идут в нужном нам конструкторе. Используя метод getConstructor(), в качестве параметра которому мы передаем наш массив, получаем экземпляр класса Conctructor. Если у Conctructor вызвать метод newInstance(args), то мы получим требуемый класс.

Осталось разобраться с последней проблемой, в решении которой нам поможет паттерн Factory Method.

Фабрика команд

Суть паттерна Factory Method заключается в том, что его реализация позволяет создавать экземпляры конкретных объектов, причем в этом случае сохраняется зависимость от абстрактных интерфейсов. Следовательно, данный шаблон в значительной мере может пригодиться в процессе активной разработки приложений, при которой конкретные классы обладают высоким уровнем изменчивости, что и требуется для решения последней задачи [2].

Если мы применим идею паттерна Factory Method для создания фабрики команд к нашему web-приложению, оно приобретет свойство

расширяемости за счет слабосвязанности классов (см. **Рис. 2**), т.е. класс AddressBookServlet напрямую не будет связан с конкретными командами, а будет взаимодействовать с ними посредством Command и CommandFactory.

Ниже представлен ключевой метод getCommand() класса CommanFactory, в котором согласно параметру name извлекаются настройки из файла конфигурации web-приложения и создается экземпляр необходимой команды (метод реализован не оптимально):

```
public Command getCommand(String name, HttpServletRequest
aRequest,
                HttpServletResponse aResponse) {
  if (name != null) {
    if (config.getInitParameter(name) != null) {
      try {
        String key = config.getInitParameter(name);
        Class cmd1 = Class.forName(config.getInitParameter(key +
            "Command"));
        try {
          Class[] parameterTypes = new Class[6];
          parameterTypes[0] = ServletContext.class;
          parameterTypes[1] = HttpServletRequest.class;
          parameterTypes[2] = HttpServletResponse.class;
          parameterTypes[3] = String.class;
          parameterTypes[4] = String.class;
          parameterTypes[5] = String.class;

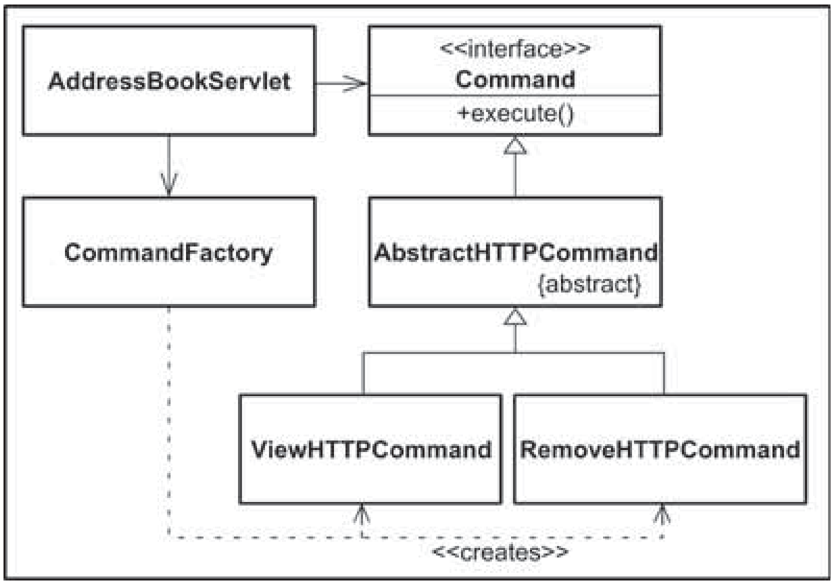
          Constructor constructor =
                cmd1.getConstructor(parameterTypes);
          Object[] args = new Object[6];
          args[0] = sc;
          args[1] = aRequest;
          args[2] = aResponse;
          args[3] = config.getInitParameter(key + "View");
          args[4] = config.getInitParameter(key + "Result");
          args[5] = config.getInitParameter(key + "Error");

          Command cmd = (Command) constructor.
newInstance(args);
          return cmd;
        } catch (Exception e) {
          e.printStackTrace();
          return null;
        }
      } catch (ClassNotFoundException e) {
        e.printStackTrace();
        return null;
      }}}
  return null;}
```

Добавим последний штрих в web-приложение – изменим метод handle класса AddressBookServlet, чтобы он работал с созданной фабрикой:

```
Command cmd;
CommandFactory cf = new CommandFactory(this.getServletConfig(),
                this.getServletContext());
cmd = cf.getCommand(
aRequest.getRequestURI().substring(aRequest.getContextPath().length()),
aRequest, aResponse);
if (cmd == null) {
  cmd = cf.getCommand("/", aRequest, aResponse);
}
try {
  cmd.execute();
} catch (Exception e) {
  // oopst...
  e.printStackTrace();
}
```

Предложенная реализация закрывает для пользователя вызов не описанных в конфигурации адресов.



› Рис. 2. Диаграмма классов, применение фабрики команд.

Что дальше?

Основной целью этой статьи было знакомство с паттернами Command и Factory Method, с чем мы неплохо справились, а заодно провели модернизацию web-приложения.

Создание экземпляров команд на основе файла конфигурации web-приложения, наверняка, не самая хорошая идея: более правильно выносить их настройку в отдельный файл (который кстати, можно будет менять из самого приложения, необходимо лишь реализовать соответствующие функции). Одно из улучшений, которое можно произвести – это изменить сами команды: сделать их stateless, то есть не хранящими внутри себя сведения о состоянии. Это улучшение позволило бы создать экземпляры команд один раз и использовать их и по мере необходимости.

Отметим, что применяя предложенные методы, можно создать приложение, модификация которого возможна без остановки его выполнения, так сказать, «на лету», но это уже тема тема отдельной статьи. LXF

Литература

1 Тейт, Б. Горький вкус Java: Библиотека программиста. – СПб: Питер, 2003. – 333 с.

2 Мартин, Р.С. Быстрая разработка программ: принципы, примеры, практика. – М.: Издательский дом «Вильямс», 2004. – 752 с.: ил.

» **Через месяц** Мы приблизим web-приложения к обычному пользователю при помощи технологий AJAX.

Начала

ЧАСТЬ 11 Завершая этот длинный цикл статей, **Евгений Балдин** посмотрит на *LaTeX* глазами профессионального программиста.

Когда придумываешь что-то сам,
высок шанс ничего не придумать.
Но когда живёшь чужим умом,
уж точно ничего не придумаешь.
Никогда не делай того, что делают другие.
Это на 100% обрекает на неудачу.

Герш Ицкович Будкер

LaTeX не просто даёт возможность набирать текст – он позволяет его программировать, а следовательно, перекладывать часть своей работы на компьютер. Привычка думать – одна из самых необычных особенностей разумного человека. Она позволяет экономить силы и время.

Создаём свои ...

...команды, окружения и прочее. Возникшая в процессе набора простенькая надоедливая проблема наверняка решена, и не один раз. С другой стороны, при нарастающей квалификации бывает проще изобрести этот велосипед заново в удобной на текущий момент форме, например:

```
\newcommand{\ee}{\ensuremath{e^{+}e^{-}}\xspace}
```

Новые команды часто создаются для комбинаций, используемых исключительно в математическом окружении. Команда \ensuremath обеспечивает это окружение независимо от текущего режима:

```
\(J/\psi\to\ee\) является одним из
подвидов \ee-рассеяния.
```

$J/\psi \to e^+e^-$ является одним из подвидов e^+e^--рассеяния.

› Рис. 1.

Команда \xspace из одноимённого пакета добавляет в конце команды пробел в случае, если за ней не следует знаков препинания, то есть избавляет от необходимости самому вставлять явный пробел после команды.

Имеются три основные структуры, которые позволяют создавать свои или переименовать уже имеющиеся макросы:

```
\newcommand{«команда»}[N][«зн. по ум.»]{«определение»}
\renewcommand{«команда»}[N][«зн. по ум.»]{«определение»}
\providecommand{«команда»}[N][«зн. по ум.»]{«определение»}
```

\newcommand определяет новую команду. Если такая команда уже существует, то при компиляции генерируется ошибка. \renewcommand переопределяет уже существующую команду. В свою очередь, \providecommand создаёт новую команду, если на момент описания такой команды не было, и ничего не делает в противном случае.

В каждом из этих макросов есть два обязательных параметра: это имя команды и её описание. Если команде необходимо передать параметр/параметры, то первый необязательный аргумент (N) должен принять значение от одного (1) до девяти (9). В LXF85 (*см. прилагаемый диск*) обсуждался макрос для дублирования знака в формуле при переносе её на следующую строку (\(a+b\hm{=}c\)):

```
\newcommand*{\hm}[1]{#1\nobreak\discretionary{}%
 {\hbox{$\mathsurround=0pt #1$}}{}}
```

Вместо знака решётки (#) с цифрой после него при компиляции макроса подставляется соответствующий параметр. В данном случае параметр был только один, и можно сказать, что его значение сохраняется в «переменной» #1.

Звёздочка (*) в конце макроса \newcommand налагает на передаваемый параметр команды \hm дополнительное условие: в нем не должно быть пустых строк и команды \par. В некоторых случаях это упрощает отладку кода.

Наличие второго необязательного параметра в макросах определения новых команд позволяет определить первый параметр создаваемой команды как параметр по умолчанию:

```
\newcommand{\exmpl}[1][умолчанию]%
                  {<<значение по #1>>}
Сравните \exmpl{} и \exmpl[требованию].
```

Сравните «значение по умолчанию» и «значение по требованию».

› Рис. 2.

Для определения нового окружения используется команда \newenvironment, например:

```
\newenvironment{outlined}{\hrule\begin{center}}%
                 {\end{center}\smallskip\hrule}
\begin{outlined}
  Выделенный текст.
\end{outlined}
```

Выделенный текст.

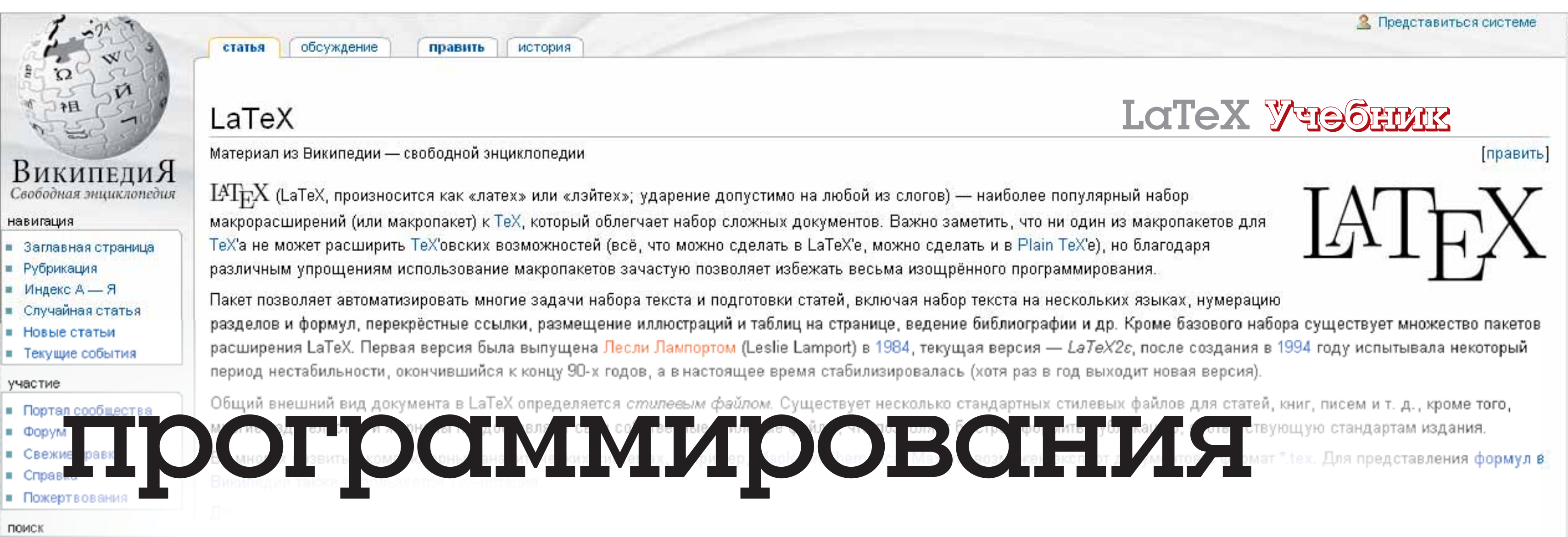
› Рис. 3.

Формальное описание этой команды похоже на описание \newcommand:

```
\newenvironment{«окружение»}[N][«зн. по ум.»]%
                  {«код, открывающий окружение»}%
                  {«код, закрывающий окружение»}
```

Точно так же, как и в случае \newcommand, созданному окружению

Месяц назад Мы разобрались со структурой книжных таблиц и их версткой в *LaTeX*.



LaTeX Учебник программирования

можно передавать параметры. Подставлять параметры можно только в коде, открывающем окружение. Кроме создания нового окружения, можно также переопределять уже имеющиеся с помощью аналогичной команды \renewenvironment.

В разделе, посвящённом описанию презентационного класса beamer (LXF85), упоминалось ещё об одной возможности создавать новые именованные окружения с помощью команды \newtheorem:

```
\newtheorem{Texmpl}{Пример}
\begin{Texmpl}[Теорема Пифагора]\label{th:1}
  Пифагоровы штаны во все стороны равны.
\end{Texmpl}

\begin{Texmpl}\label{th:2}
  Мудрость ограничена, а глупость бесконечна.
\end{Texmpl}

Можем сослаться первую теорему:~\ref{th:1},
а можно и на вторую:~\ref{th:2}
```

Пример 1 (Теорема Пифагора). *Пифагоровы штаны во все стороны равны.*

Пример 2. *Мудрость ограничена, а глупость бесконечна.*

Можем сослаться первую теорему: 1, а можно и на вторую: 2

› Рис. 4.

Команда \newtheorem имеет две формы:

```
\newtheorem{«теорема»}[«существующая теорема»]{«заголовок»}
\newtheorem{«теорема»}{«заголовок»}[«имя счётчика»]
```

Каждая из форм имеет по два соответствующих обязательных аргумента и по одному необязательному. В первом случае это имя уже существующей теоремы, с которой следует иметь совместную нумерацию. Во втором случае в качестве необязательного параметра передаётся имя уже существующего счётчика, на основе которого строится нумерация. О том, что такое счётчики и как их определять, речь пойдёт далее.

Счётчики и другие переменные

«Другие переменные» уже обсуждались в разделе Определённые «размеры» и переменные «длины» (LXF89). Операции с этими переменными выполнялись с помощью команд \newlength, \setlength и \addtolength. Аналогично, в *LaTeX* представлена и целочисленная арифметика с использованием счётчиков в качестве переменных:

```
\newcounter{MyCount}\setcounter{MyCount}{5}
Значение MyCount равно \arabic{MyCount},
или~\alph{MyCount}, или  \Asbuk{MyCount}.\par
\addtocounter{MyCount}{1550}
\arabic{MyCount} эквивалентно \Roman{MyCount}.
```

Значение MyCount равно 5, или e, или Д.

1555 эквивалентно MDLV.

› Рис. 5.

Новый счётчик создаётся с помощью команды \newcounter. При создании он инициализируется нулём. Создание счётчика является глобальной операцией, то есть при компиляции информация о нём не исчезнет, даже если новый счётчик был определён внутри окружения. Для присвоения счётчику другого значения используется команда \setnewcounter, а для изменения на какое-то определённое число – \addtocounter.

В отличие от длин, основная роль которых – помнить размеры определённого бокса, счётчики используются для отображения какой-либо структурной информации. Поэтому особое внимание уделяется представлению значения счётчика в тексте. Чтобы просто отобразить численное значение счётчика с помощью арабских цифр, используется команда \arabic{счётчик}. Для римской числовой нотации необходимо воспользоваться командой \Roman и \roman – заглавные и строчные буквы соответственно. Счётчик может быть также представлен буквой алфавита: \alph – латинская строчная, \asbuk – кириллическая строчная и \Asbuk – кириллическая заглавная.

В стандартных классах уже определён набор счётчиков, в которых хранятся номера страницы (счётчик page), раздела (соответственно, счётчики part, chapter, section, subsection, subsubsection и т.д.), подстрочного примечания (счётчик footnote), плавающих окружений (счётчики figure и table) и формул (equation). При создании счётчика также автоматически создаётся команда с префиксом \the перед именем счётчика. Вызов такой команды выводит значение счётчика. При выводе номера раздела, плавающего объекта, уравнения и тому подобного используются именно такого рода команды, поэтому, переопределив \the-команду, можно немного изменить стиль, например, следующая команда предписывает в дальнейшем маркировать все страницы в римском стиле:

```
\renewcommand{\thepage}{\Roman{page}}
```

На базе счётчиков можно организовывать иерархические структуры, то есть можно указывать зависимости:

```
\newcounter{Main}\addtocounter{Main}{10}
\newcounter{Dep}[Main]\addtocounter{Dep}{10}
Было: \theMain.\theDep\par
\stepcounter{Main}
Стало: \theMain.\theDep
```

Было: 10.10

Стало: 11.0

› Рис. 6.

При создании нового счётчика можно создать связь с уже существующим, указав имя существующего счётчика в качестве необязательного параметра. В примере выше счётчик Dep зависит от счётчика Main. Эта связь проявляется в том, что при увеличении значения базового счётчика (Main) на единицу с помощью команды \stepcounter подчинённый счётчик (Dep) обнуляется. Обычно новый счётчик устанавливают в подчинение счётчикам разделов (section).

Команда \refstepcounter{счётчик} отличается от \stepcounter тем, что помимо обнуления всех зависимых счётчиков, \refstepcounter определяет значение, выводимое командой ссылки \ref, как текст, создаваемый \the-командой:

```
% окружение "Задача"
\newcounter{Problem}[section]
\renewcommand{\theProblem}{\thesection.\arabic{Problem}}
\newenvironment{Problem}[0]{%
 \par\refstepcounter{Problem}%
 \theProblem\,}%
{\par}%
```

Здесь определено окружение Problem и одноимённый счётчик. Счётчик Problem зависит от счётчика раздела. Вывод счётчика \theProblem переопределён как номер раздела, за которым следует уже



» сам счётчик. Внутри окружения счётчик Problem увеличивается на единицу с помощью команды \refstepcounter{счётчик}. Результат использования нового окружения представлен в следующем примере:

```
\begin{Problem}\label{ex:1}
  Задача раз
\end{Problem}
\begin{Problem}\label{ex:2}
  Задача два
 \end{Problem}
Ссылки на раз~\ref{ex:1} и два~\ref{ex:2}.
```

11.2.1 Задача раз

11.2.2 Задача два Ссылки на раз 11.2.1 и два 11.2.2.

› **Рис. 7.**

При работе с переменными *LaTeX* также могут помочь следующие пакеты:

» *calc* Макропакет из коллекции tools для арифметических вычислений, уже упоминавшийся в разделе calc (LXF89). Этот пакет переопределяет команды типа \newcounter так, что в них можно использовать арифметические выражения, хотя и с некоторыми ограничениями. Подробности в файле **calc.pdf**.

» *ifthen* Макропакет, в котором определены команды условного перехода \ifthenelse и цикла \whiledo. Подробности в файле **ifthen.pdf**. Также можно присмотреться к усовершенствованной версии этого пакета *xifthen*.

» *fmtcount* Представляет различные форматы (двоичный, восьмеричный, шестнадцатеричный и т.д.) отображения счётчиков (**fmtcount.pdf**).

» *multido* Определяет оператор цикла \multido (**multido.pdf**).

» *tokenizer* Позволяет разбивать текстовые списки на элементы (**tokenizer.pdf**).

» *totpages* Даёт возможность узнать число страниц в документе и тому подобную информацию (**totpages.pdf**).

» *xkeyval* Улучшенная версия пакета keyval, который позволяет передавать/принимать в качестве параметров пары значений «key=value» (**xkeyval.pdf**).

Создаём свой пакет

Предположим, что вы уже владеете искусством программирования в среде *LaTeX*. Чтобы распространить свои наработки, следует организовать исходные тексты в удобном для дальнейшей поддержки, передаче и установке виде. Можно и не стараться, если вас не интересует конечный результат, но знать, как правильно устроен пакет, полезно и новичку, так как эффективное обучение программированию напрямую связано с изучением уже существующего кода.

В *LaTeX*-сообществе принято распространять свои пакеты и документацию к ним в виде автономных файлов с расширением dtx (DTX-файлы). Для автоматической установки пакетов используются инструкции, записанные в файлах с расширением ins (INS-файлы). Для более подробной информации следует обратиться к инструкции «How to Package Your *LaTeX* Package», созданной Скотом Пакиным [Scott Pakin]. Файл **dtxtut.pdf**, как обычно, можно найти в стандартной поставке *LaTeX* или на *CTAN*. Вместе с документацией идут файлы примеров **[c]skeleton.dtx** и **[c]skeleton.ins**.

Внимание!

Политика создания названий команд в TeX-подобной среде такова, что для новых пакетов необходимо придумывать новые команды. Это сделано для обеспечения абсолютной совместимости сверху вниз. К сожалению, подобная политика в случае бездумного использования слов может привести к «захватыванию» подходящих сочетаний. Примером правильного именования служит пакет *listings*, где вместо, казалось бы, подходящего по названию окружения listing используется lstlisting.

За работу с DTX-файлами отвечает пакет doc и сопутствующая ему утилита DOCSTRIP (файл **docstrip.pdf**). Основная идея пакета doc состоит в совмещении кода с документацией, что облегчает поддержку и развитие пакета.

Установочный INS-файл

Для извлечения кода и документации из DTX-пакета следует написать специальный установочный файл. Набор инструкций достаточно стереотипен:

```
%% Стандартный копирайт по выбору (рекомендуется LPPL/GPL)
%%
%% Первый шаг - загрузка DOCSTRIP.
\input docstrip.tex
%% Подробный отчёт о каждом шаге хорош только когда пакет
%% отлаживается.
\keepsilent
%% Директория, в которую устанавливается пакет. Имя
%% директории является относительным по отношению к
%% базовой директории $(TEXMF).
\usedir{tex/latex/{«имя пакета»}}
%% Определение преамбулы, которая вставляется во все
%% сгенерированные файлы. Обычно это информация об авторе
%% и пожелания пользователям.
\preamble
 Текст преамбулы
\endpreamble
%% Извлечение файлов пакета из dtx. Основной шаг,
%% который может повторяться несколько раз.
\generate{\file{«извлекаемый файл»}{\from{«dtx-файл»}{метка}}}
   ...
   ...
%% Информация для пользователя. Всегда что-то полезно сказать
%% после установки.
\obeyspaces
\Msg{*******************************************************
**}
\Msg{*      Здорово, что Вы поставили этот пакет.         *}
\Msg{*    Прочитайте документацию перед использованием!    *}
\Msg{*******************************************************
**}
%% Метка конца установочного файла.
\endbatchfile
```

Пакетный DTX-файл

Пакетный DTX-файл содержит в себе и код с комментариями, и текст описания пакета. Структура DTX-файла позволяет получить печатную документацию после прогона через *LaTeX*. Код с комментариями тоже может стать частью документации. Это шаг по направлению к «грамотному программированию» [*literate programming*].

Наличие комментариев в коде заставляет повторять процедуру компиляции дважды. В первый раз отрабатывается *LaTeX*-код, а затем комментарии. Во втором случае знак % перед комментарием игнорируется, и текст комментария передаётся на вход *LaTeX*, если он (комментарий) не окружён командами \iffalse-\fi.

Пролог

В начале следует, естественно, добавить информацию об авторе:

```
%\iffalse meta-comment
% Этот текст не обрабатывается LaTeX'ом. Слово meta-comment
%добавлено просто для удобства чтения кода человеком и
%означает, что этот текст предназначен именно для него (человека).
%\fi
```

В ins-файле в команде \generate использовался параметр «метка». Он говорит DocStrip, что следует выбирать строки, которые следуют за комментарием и конструкцией <метка> или между тэгами <*метка> и

</метка>. Далее идёт код заголовка пакета, соответствующего метке «метка»:

```
% \iffalse
%<метка> \NeedsTeXFormat{LaTeX2e}
%<метка> \ProvidesPackage{«имя пакета»}
%<метка>    [<ГГГГ>/<ММ>/<ДД> v<версия> <краткое описание>]
```

Строчку «<ГГГГ>/<ММ>/<ДД> v<версия> <краткое описание>» нужно заменить на дату, версию и краткое описание, соответственно.

Закончить пролог необходимо следующими словами, создающими основную документацию:

```
%<*driver>
\documentclass{ltxdoc}
\usepackage{«имя пакета»}
\begin{document}
 \DocInput{«dtx-файл»}
\end{document}
%</driver>
% \fi
```

Это единственная часть, относящаяся к документации, которая не начинается со знака комментария (%).

В прологе можно указывать ещё некоторое количество инструкций, уточняющих формат создаваемой документации.

Пользовательская документация

Прежде всего, следует учесть, что подавляющий объём описаний для пакетов *LaTeX* сделан на английском языке. Для этого есть довольно веские основания, связанные с размером англоязычной аудитории.

Написание документации для DTX-пакета ничем не отличается от написания обычного *LaTeX*-документа, за тем исключением, что не следует забывать о знаке комментария (%) в начале строки:

```
% \title{Пакет \textsf{«имя пакета»}}
% \author{«Ваше имя» \\ \texttt{«Ваш e-mail»}}
% \maketitle
% текст документации
```

К стандартным *LaTeX*-командам секционирования уровня paragraph добавляются \DescribeMacro{макрос} и \DescribeEnv{окружение}.

Код с комментариями

Даже если допустить, что лучшая документация для программиста – это сам код, то для нормального человека описательный текст все равно будет предпочтительнее. Проблема совмещения кода и описания является основной причиной возникновения «грамотного программирования».

Код обычно начинается сразу же за пользовательской документацией:

```
%\StopEventually{\PrintIndex}
%% Описание окружения.
%\begin{environment}{«имя окружения»}
%% Аналогично, существует окружение macros, для описания
%% новых команд.
%    \begin{macrocode}
 «Здесь идёт код, вида:»
\newenvironment{«имя окружения»}{начало}{окончание}
%    \end{macrocode}
%\end{environment}
 …
%\Finale
\endinput
```

Команда \StopEventually{} отмечает начало кода и принимает в качестве параметра команду, которую следует выполнить в конце документации – например, распечатать алфавитный указатель \PrintIndex.

Любой код следует обрамлять с помощью окружения macrocode. Это позволит включить его в печатную документацию. Есть две особенности для этого окружения, которые следует учитывать:

» Между % и \begin{macrocode} должно быть ровно четыре (4) пробела. Аналогичное правило действует и для \end{macrocode},

» Внутри этого окружения не должно быть текста, начинающегося с %.

Внутри окружений environment и macros может быть несколько вставок кода и текста.

Пакетирование

Часто *LaTeX*-пакеты распространяются в виде одного DTX-файла. Существует способ включить установочный INS-файл в файл пакета:

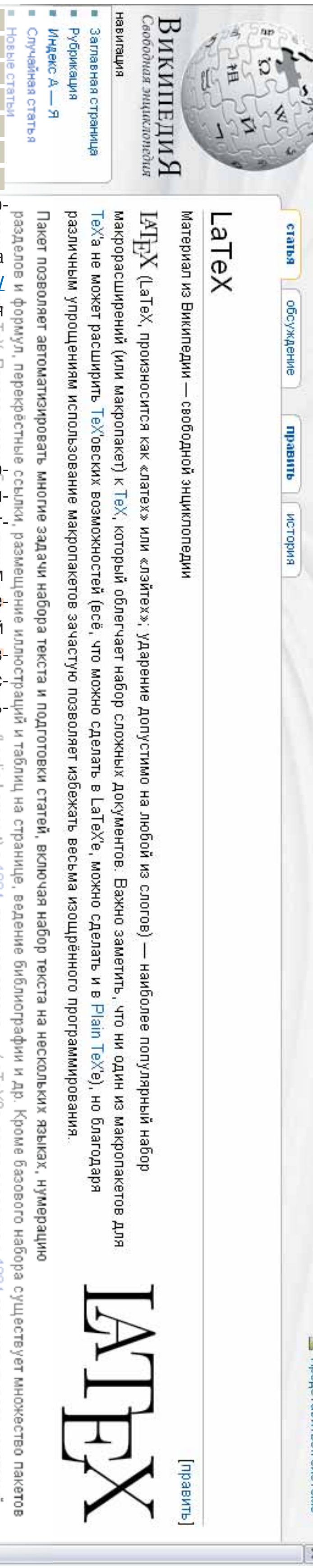
```
%<*batchfile>
\begingroup

 «Содержание ins-файла»

\endgroup
%</batchfile>
```

Следует только убрать заключительную команду \endbatchfile, чтобы *LaTeX* мог скомпилировать остальное.

Всё. Для распространения свой пакет лучше всего поместить на CTAN. Для выгрузки следует обратиться к ресурсу http://www.ctan.org/upload. Всегда необходим краткий README с описанием. Собранная документация в виде PDF-файла также является хорошим тоном.



Напутствие

Документируйте каждый шаг. Пишите как можно больше качественного текста, так как его мало не бывает. Живучесть программы определяется не только кодом, но и описанием. «Светлое будущее» за грамотным программированием.

LaTeX-цикл в *Linux Format* подошёл к концу. Честно говоря, я сам за это время узнал много нового для себя. Надеюсь, мне удалось поделиться этими знаниями с вами. В этой информации нет никакой чёрной магии – всё просто и логично, и эта информация полезна, так как позволяет автоматизировать одно из самых сложных ремёсел человеческой цивилизации – создание книг. Пишите тексты, большие и маленькие: они не пропадут. LXF

Защити

ЧАСТЬ 2 Настало время подкрепить полученные знания практикой. **Андрей Боровский** расскажет как получить цифровой сертификат и использовать его в популярных открытых приложениях.

In Web we trust.
Девиз сетей доверия.

Как получить сертификат

Выбор способа получения сертификатов зависит от целей их использования. Если сертификаты используются для безопасной переписки несколькими людьми, которые знакомы между собой «в реале» (сотрудники небольшой компании, например), ничто не мешает им воспользоваться «самопальными» сертификатами, сгенерированными с помощью утилиты *openssl* (как это делается, я расскажу чуть ниже). Если сертификат необходим для обмена сообщениями с незнакомыми людьми в глобальной Сети, следует обратиться в один из отечественных или зарубежных удостоверяющих центров (УЦ). Наконец, если сертификат требуется для взаимодействия с госструктурами РФ, его следует получать в одном из УЦ, утвержденных нашими государственными органами (*см. врезку «Особенности национальной сертификации»*).

Получение бесплатного сертификата

Удостоверяющие центры работают на коммерческой основе, но получить сертификат, удостоверенный УЦ, можно и бесплатно (такая услуга предоставляется частным лицам и некоммерческим проектам многими крупными УЦ). Возможности применения бесплатных сертификатов несколько ограничены по сравнению с коммерческими – дело, разумеется, не в самих сертификатах, а в уровне услуг, предоставляемых УЦ. Обычно бесплатные сертификаты можно получать в онлайн-режиме, откуда следует, что по умолчанию бесплатный сертификат связывает с открытым ключом только те данные, которые могут быть проверены онлайн (то есть адрес электронной почты или доменное имя). Например, для удостоверения того, что некий адрес электронной почты действительно принадлежит лицу, обратившемуся за сертификатом, на этот адрес высылается письмо с запросом. Чтобы получить право вписать в сертификат дополнительные сведения, которые нельзя проверить по Сети, придется предпринять дополнительные усилия (*см. врезку «Как попасть в сети доверия»*).

Наибольшей известностью среди УЦ, предоставляющих часть услуг бесплатно, пользуются три: Thawte Consulting, CAcert и Comodo Group. У каждого из этих УЦ есть свои достоинства и недостатки.

Компания Comodo Group (известная многим благодаря своему бесплатному брандмауэру для Windows) использует для распространения сертификатов возможности браузера *Internet Explorer*, поэтому ее услуги для нас не подходят (разумеется, установив сертификат в браузере Windows, его потом можно «перетащить» в любую другую программу, в том числе и на платформу Linux, но мы, конечно, этого делать не будем).

УЦ CAcert поддерживает и Windows, и Linux, и вообще предоставляет самый широкий спектр бесплатных услуг, но корневые сертификаты (КС) от CAcert не предустановлены в популярных браузерах и почтовых клиентах (по крайней мере, на данный момент), а это значит, что для удостоверения вашего сертификата CAcert вашему контрагенту придется самому устанавливать КС от CACert. Далеко не каждый пользователь согласится что-то там устанавливать из Интернета, тем более что сообщения браузеров, не обнаруживших КС, выглядят пугающе. Например, *IE 7* вообще советует уносить ноги с сайта CAcert (**Рис. 1**).

УЦ Thawte (основанный, кстати, небезызвестным Марком Шаттлвортом и проданный затем компании VeriSign) – большое имя в мире цифровых сертификатов – предоставляет бесплатные сертификаты для удостоверения электронной почты (но не для SSL). Этот УЦ позаботился о том, чтобы его сертификаты можно было установить в любой системе, кроме того, КС от Thawte предустановлены практически во всех программах, которые работают с цифровыми сертификатами. Если сертификат нужен вам для подписи и шифрования частной почты, бесплатные сертификаты Thawte – лучший выбор. Недостатком Thawte можно назвать то, что при выдаче бесплатных сертификатов не поддерживаются самостоятельно сгенерированные запросы на получение сертификатов (см. ниже), вдобавок Thawte не поддерживает установку сертификатов в *Konqueror*. Это означает, что для установки бесплатных сертификатов Thawte нужно использовать *Firefox* или *Opera*.

Особенности национальной сертификации

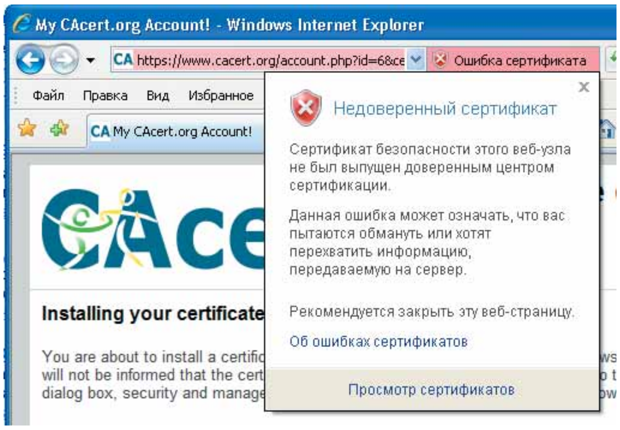
По вполне понятным причинам государства контролируют (по крайней мере, отчасти) применение систем шифрования своими гражданами, и наше государство не является исключением. Если я правильно понимаю соответствующие законы нашей страны (а я не претендую на абсолютно правильное их понимание), пользоваться программами, прошедшими государственную сертификацию, обязательно только внутри государственных органов РФ и при взаимодействии с таковыми. В остальных случаях пользователь вправе защищать свои данные теми средствами, которыми он располагает. Удостоверяющие центры в нашей стране есть, и их немало. В числе предлагаемых ими алгоритмов и протоколов электронной подписи можно найти как определенные ГОСТами, так и описанные в различных RFC и FIPS.

В качестве материала для размышлений (а возможно, и для медитации) привожу Статью N 5, пункт 2, Федерального закона от 10 января 2002 года «Об электронной цифровой подписи»:
«При создании ключей электронных цифровых подписей для использования в информационной системе общего пользования должны применяться только сертифицированные средства электронной цифровой подписи. Возмещение убытков, причиненных в связи с созданием ключей электронных цифровых подписей несертифицированными средствами электронной цифровой подписи, может быть возложено на создателей и распространителей этих средств в соответствии с законодательством Российской Федерации».

С текстами законов РФ, регламентирующих использование электронной цифровой подписи и некоторых других средств, можно ознакомиться, например, на сайте www.ekey.ru/lib/1.

Месяц назад Мы разобрались с «теорией цифровых сертификатов».

свои данные



› **Рис. 1. «Недоверенный» сертификат.**

Получение сертификата по Сети

Существует две схемы получения сертификатов онлайн. В первом варианте и сертификат, и пара ключей генерируются вашим браузером и отправляются на сайт УЦ. Когда ваш сертификат появится на сайте УЦ, вы сможете установить сертификат (и прилагающийся секретный ключ) в вашем браузере. Чтобы импортировать сертификат и секретный ключ (СК) в *Firefox*, достаточно щелкнуть по ссылке, указывающей на сертификат на сайте УЦ. Учтите, что хотя ваш сертификат будет храниться на сервере УЦ, СК к сертификату вы можете получить только один раз, после чего УЦ его «забудет» (а без СК сертификат бесполезен).

Второй вариант – создать запрос на получение сертификата с помощью утилиты *openssl* и отправить его на сайт УЦ (похоже, что *Konqueror* не умеет генерировать правильные запросы на получение сертификата, так что если вы не пользуетесь *Firefox* или *Opera*, этот вариант – для вас). Запрос на получение сертификата представляет собой документ, подписанный с использованием секретного ключа из новой пары ключей. Чтобы создать запрос на получение сертификата, нужно сгенерировать ключи:

```
openssl genrsa -out client.key 2048
```

Секретный ключ сохраняется в файле **klient.key** (там же, на самом деле, сохраняется и открытый ключ, без которого СК тоже бесполезен). Последнее число в строке команды – длина ключа в битах, определяющая его стойкость.

Запрос на получение сертификата (файл **csr**) создается с помощью команды:

```
openssl req -new -key client.key -out client.csr
```

Во время создания запроса на получение сертификата утилита попросит вас ввести несколько параметров, важнейшими из которых (коль скоро речь идет о бесплатных сертификатах) являются Common Name и Email Address (для сертификата, удостоверяющего почтовый адрес, в обоих параметрах указываем свой e-mail). Содержимое файла **client.csr** следует передать удостоверяющему центру. Сертификат, который выпустит УЦ, в этом случае, скорее всего, будет доступен вам в виде файла (в формате PEM или CRT). Сохраните этот файл на диске, и помните, что без файла, содержащего секретный ключ, файл сертификата не имеет особой ценности.

Файлы сертификатов X.509

Получение, выдача, передача и резервное копирование сертификатов и секретных ключей сопровождаются сохранением их данных в специальных файлах. Чаще всего для этого используются файлы со следующими расширениями:

***.cer** – сертификат, сохраненный в стандарте CER. Может включать сертификат, секретный ключ, путь сертификации.

***.der** – сертификат, сохраненный в стандарте DER. Может включать сертификат, секретный ключ, путь сертификации.

***.crt** – файл сертификата в формате CER, DER или Netscape.

***.pem** – сертификат в кодировке Base64. Может также включать полный путь удостоверения сертификата и секретный ключ.

***.p8** – файл, содержащий секретный ключ, защищенный по стандарту PKCS#8.

***.p12** (в Windows используется расширение ***.pfx**) – файл сертификата, защищенный по стандарту PKCS#12. Может включать сертификат, секретный ключ, путь сертификации.

Установка сертификатов в программах Linux

Многими популярными программами Linux поддерживаются цифровые сертификаты X.509. Мы рассмотрим их установку и использование в программах *KMail*, *Thunderbird* и в пакете *OpenOffice.org*. Чтобы иметь возможность подписывать собственные сообщения, вы должны установить в программах секретный ключ и сертификат, удостоверяющий соответствующий ему открытый ключ. Для шифрования сообщений, предназначенных другим людям, достаточно установить сертификаты ваших корреспондентов (из этих сертификатов почтовая программа извлечет необходимые для шифрования открытые ключи). Сертификат вашего корреспондента вы можете получить, например, вместе с письмом, содержащим его электронную подпись.

Для установки и хранения сертификатов вместе с секретными ключами мы воспользуемся форматом PKCS#12 (*см. врезку «Файлы сертификатов X.509»*). Импортировав личный сертификат и секретный ключ с сайта УЦ в браузер *Firefox*, вы можете экспортировать его в виде пакета PKCS#12 с помощью менеджера сертификатов. В окне Настройки Firefox перейдите в группу настроек Дополнительно, откройте вкладку Шифрование и щелкните кнопку Просмотр сертификатов.

В открывшемся окне менеджера сертификатов выберите ваш сертификат и щелкните кнопку Резервная копия. Вам будет предложено указать пароль для защиты секретного ключа, после чего файл PKCS#12 можно будет сохранить на диске. С помощью менеджера сертификатов вы можете также импортировать пакет PKCS#12 в браузер *Firefox*.

Если вы получили сертификат и секретный ключ в формате PEM, можете воспользоваться утилитой *openssl* для преобразования его в PKCS#12. В окне консоли скомандуйте (мы сохраняем пакет PKCS#12 под именем **cert.p12**):

```
openssl pkcs12 -in cert.pem -export -out cert.p12
```

где **cert.pem** – файл с сертификатом и ключом в формате PEM, а **cert.p12** – новый файл в формате PKCS#12. »

» При наличии сертификата и секретного ключа в формате ***.crt**, вы также можете преобразовать их в формате PKCS#12 с помощью *openssl*:

```
openssl pkcs12 -in cert.crt -export -out cert.p12
```

Если сертификат и секретный ключ хранятся в разных файлах (например, вы сгенерировали ключи сами, а потом создали запрос на получение сертификата), команда создания пакета PKCS#12 будет выглядеть иначе:

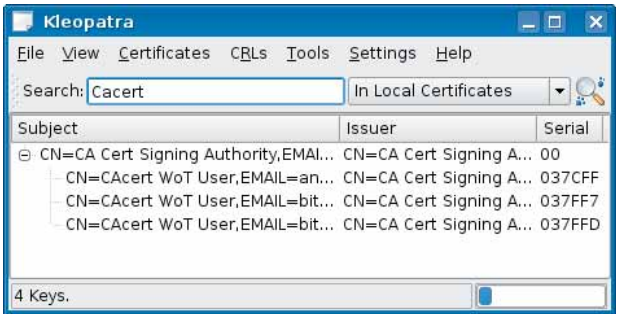
```
openssl pkcs12 -export -in cert.pem -inkey client.key -out cert.p12
```

Здесь **cert.pem** – файл сертификата, загруженный с сайта УЦ, а **client.key** – файл секретного ключа, который вы создали для генерации запроса на получение сертификата.

Во всех случаях утилита *openssl* предложит вам указать пароль для защиты пакета PKCS#12.

Установка сертификата в KDE/KMail

Чтобы импортировать пакеты сертификатов и секретных ключей из файлов PKCS#12 (и других форматов) в KDE, достаточно открыть файл сертификата в менеджере *Konqueror* и щелкнуть кнопку Import (при этом будет запрошен пароль, защищающий пакет PKCS#12). Импортировать сертификаты в KDE можно также с помощью программы *Kleopatra* (**Рис. 2**).



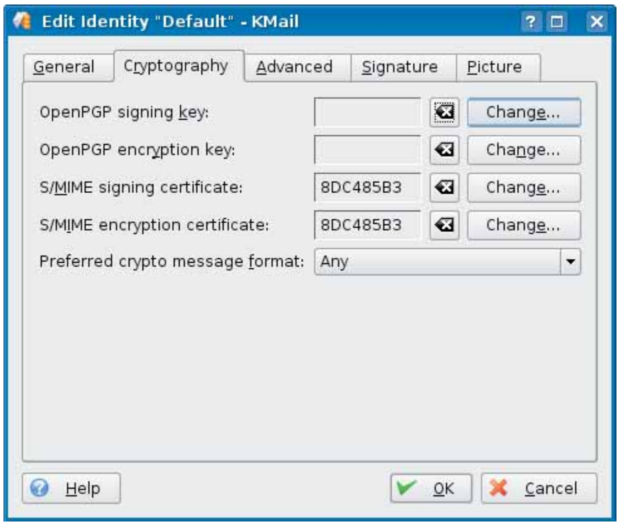
› **Рис. 2. *Kleopatra*.**

Программа потребует пароль, которым защищен пакет PKCS#12, затем предложит указать дважды новый пароль для сохранения данных в настройках KDE. Установить сертификат в KDE можно также с помощью центра управления рабочим столом KDE. В окне центра управления перейдите в группу Security & Privacy -> Crypto, откройте вкладку Your Certificates и воспользуйтесь кнопкой Import для импортирования сертификата из файла. При этом программа спросит, должен ли данный сертификат быть доступен *KMail*.

Импортировав сертификат в KDE, вы можете настроить клиент *KMail* для работы с ним. В окне Edit Identity перейдите на вкладку Cryptography (**Рис. 3**) и выберите сертификат для подписи сообщений по протоколу S/MIME (S/MIME signing certificate) и шифрования сообщений (S/MIME encryption certificate). При установке сертификата для электронной подписи в *KMail* нужно следить за тем, чтобы почтовый адрес, включенный в сертификат, совпадал с адресом, для которого выполняется настройка, иначе почтовый клиент адресата выдаст сообщение о подлоге. Если вы подписываете письмо электронной подписью, *KMail* потребует у вас пароль, которым защищен соответствующий секретный ключ.

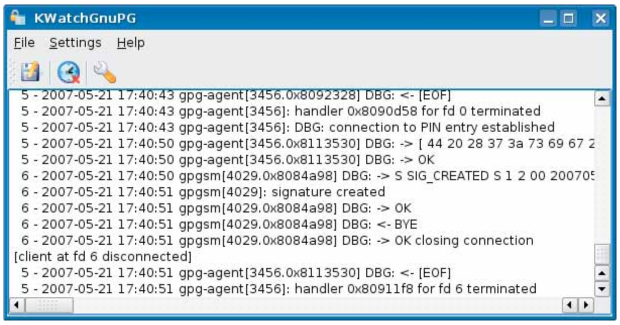
Ситуация несколько усложняется, если вы устанавливаете личный сертификат, заверенный корневым сертификатом, отсутствующим в вашей системе. Чтобы установить в KDE, например, КС от CAcert, его нужно сначала импортировать так же, как и личный сертификат. После этого необходимо добавить цифровой отпечаток SHA-1 корневого сертификата в список доверенных сертификатов gpgsm (список отпечатков доверенных сертификатов хранится в файле **~/.gnupg/trustlist.txt**). Получить отпечатки SHA-1 всех установленных сертификатов вы можете с помощью команды

```
gpgsm --fingerprint
```



› **Рис. 3. Окно настройки шифрования в *Kmail*.**

Консольные утилиты gpgsm и gpg-agent выполняют все действия, связанные с обработкой сертификатов X.509 и связанных с ними ключей в KDE, программа *KMail* запускает их как внешние процессы. Посмотреть сообщения, выдаваемые этими утилитами, и выявить причину возможных сбоев можно с помощью программы *KWatchGnuPG* (**Рис. 4**).



› **Рис. 4. Окно *KwatchGnuPG*.**

Установка сертификата в Thunderbird

Установка сертификата и секретного ключа из файлов PKCS#12 в *Thunderbird* выполняется с помощью менеджера сертификатов этой программы (*Thunderbird* мог бы и сам извлекать сертификаты из хранилища *Firefox*, но он этого не делает). Откройте в *Thunderbird* окно Preferences, перейдите в группу Privacy, откройте вкладку Security и вызовите менеджер сертификатов, щелкнув кнопку View Certificates. Теперь вы можете добавить сертификат, щелкнув кнопку Import.

В процессе установки сертификата и секретного ключа из пакета PKCS#12 *Thunderbird* спросит у вас два пароля – мастер-пароль, который программа использует для защиты хранящихся в ее настройках секретных ключей, и пароль, которым защищен пакет PKCS#12. Если вы используете сертификат CAcert, вам необходимо также установить КС CAcert. (КС доступен на сайте CAcert, его следует устанавливать точно так же, как и личный сертификат). Установив КС, перейдите на вкладку Authorities в окне менеджера сертификатов, найдите в списке нужный КС, щелкните кнопку Edit и установите флажок This Certificate Can Identify Mail Users.

При отправке подписанного сообщения из *Thunderbird* следите за тем, чтобы почтовый ящик, с которого вы посылаете сообщение, совпадал с почтовым ящиком, указанным в сертификате. Так же, как и *KMail*, клиент *Thunderbird*, перед тем как подписать сообщение, запрашивает пароль, защищающий секретный ключ.

Как попасть в «сети доверия»?

Если вы хотите внести в бесплатный цифровой сертификат какие-либо сведения о себе помимо адреса электронной почты, необходимо убедить УЦ в достоверности этих сведений. Для этого вам придется лично встретиться с кем-то, кому УЦ доверяет. Даже если вы сами не можете посетить УЦ, выдающий бесплатные сертификаты, отчаиваться не стоит: вполне возможно, что УЦ уже пришел к вам. Для удостоверения сертификатов «на местах» многие УЦ создают сети доверия (Webs of Trust, WoTs). Сети доверия есть и у CAcert, и у Thawte. Сеть доверия – это группа частных лиц, которые пользуются доверием УЦ (назовем их нотариусами). Нотариусы сетей доверия участвуют в процессе удостоверения дополнительных сведений о владельцах сертификатов, а также в удостоверении новых нотариусов (в результате чего сеть доверия может расти «сама собой»).

Рассмотрим в качестве примера работу сети доверия Thawte. Чтобы получить право добавить свое имя в цифровой сертификат Thawte, необходимо встретиться лично с несколькими нотариусами Thawte из WoT и представить им по два документа с фотографиями (а также передать копии этих документов), удостоверяющих вашу личность (один из документов должен быть упомянут в вашей учетной записи Thawte).

Почему нужно встречаться с несколькими нотариусами WoT? Как и многие другие УЦ, Thawte измеряет доверие пунктами [assertion points]. Нотариусы WoT не удостоверяют вас сами, они лишь присваивают вам определенное количество пунктов доверия. Чтобы внести имя в свой сертификат Thawte, вы должны набрать 50 пунктов, а чтобы самому стать нотариусом – 100 пунктов. В то же время каждый нотариус Thawte WoT может присвоить вам от 10 до 35 пунктов, не более (данное ограничение затрудняет обретение доверия путем подкупа нотариусов, использования личных связей с нотариусами, давления на них и т.п.). Все это немного напоминает ролевую игру с пунктами «экспы». Хорошо еще, что для получения вожделенных пунктов доверия не требуется выиграть поединок на пластиковых мечах!

Нотариусы Thawte WoT есть и в нашей стране (по крайней мере, в крупных городах; более точную информацию вы можете получить, зарегистрировавшись на сайте Thawte). Некоторые нотариусы предоставляют свои услуги бесплатно, другие – за скромное вознаграждение (1–2 у.е.). Если удостоверение с помощью нотариусов вам не подходит, Thawte предлагает и другие способы идентификации (например, с помощью доверенных третьих лиц), но стоить это будет дороже. Напомню, что речь идет только о сертификатах Thawte, выдаваемых бесплатно. Полностью коммерческий сертификат Thawte можно получить, обратившись к представителю Thawte (в России, например, в компанию РБК).

Матерые пользователи OpenPGP часто смотрят свысока на цифровые сертификаты X.509. Тем не менее, именно сертификаты этого формата властвуют за пределами Linux, а потому мы должны использовать их в общении с не-линуксоидами. В конце концов, уважение к чужим традициям – важнейший принцип доброжелательного общения.

Подпись почтового сообщения с помощью openssl

Если ваша любимая почтовая программа не поддерживает сертификаты X.509 напрямую, это еще не значит, что вам нельзя ими пользоваться. С помощью утилиты *openssl* и пакета «сертификат + СК» в формате PEM (такой пакет можно создать, например, с помощью программы *Kleopatra*), вы можете подписывать почтовые сообщения, сохраненные в файле на диске:

```
openssl smime -sign -in message.eml -out message-signed.eml -signer cert.pem -noverify
```

Опция **-noverify** отключает проверку сертификата, удостоверяющего подпись. Файл **message-signed.eml** будет содержать подписанное сообщение. Другие опции *openssl* позволяют встроить эту утилиту в цепочку программ, выполняющих создание и отправку сообщений. Советую ознакомиться с документацией к этой мощной утилите.

Использование сертификатов в OpenOffice.org

Программы из пакета *OpenOffice.org* могут использовать для подписи документов ключи и сертификаты, установленные в браузере *Firefox* (OpenOffice умеет извлекать сертификаты из хранилища *Firefox* автоматически). Если ваш личный сертификат не установлен в *Firefox*, вы можете установить его с помощью менеджера сертификатов *Firefox*, описанного выше. Чтобы подписать документ в *OpenOffice.org*, необходимо выбрать команду меню File | Digital Signatures и в открывшемся окне щелкнуть кнопку Add. Будет открыт список личных сертификатов, в котором можно выбрать сертификат/ключ для подписи.

О том, что документ *OpenOffice* подписан и цифровая подпись цела, свидетельствует пиктограмма в строке состояния окна *OpenOffice*, изображающая документ с красной печатью (если изменить подписанный документ, этот значок исчезнет).

Желтый треугольник с восклицательным знаком на пиктограмме свидетельствует о том, что проверить подлинность сертификата, подтверждающего подпись, не удалось. Такое может произойти, если соответствующий КС не установлен, или для него не установлено право удостоверения сертификата, предназначенного для подписи контента.

Для исправления этой ошибки откройте окно менеджера сертификатов *Firefox* и перейдите на вкладку Центры сертификации. Если КС для вашего сертификата отсутствует, его следует установить в *Firefox*. Если нужный КС установлен, выделите его в окне менеджера сертификатов и щелкните кнопку Изменить. Установите флажки Этот сертификат может идентифицировать пользователя электронной почты и Этот сертификат может идентифицировать программное обеспечение.

Жирный восклицательный знак на пиктограмме, отображающей состояние подписи документа *OpenOffice*, свидетельствует о том, что структура подписи была нарушена (или удостоверяющий корневой сертификат не валиден). LXF



Путь

ЧАСТЬ 3 *Cinelerra* – это не *Synfig* и не *Blender*, но смонтировать в ней небольшой мультфильм вполне возможно. Особенно если вы, как и **Петр Семилетов**, предпочитаете не рисовать, а лепить...

В этой статье мы поглядим, насколько *Cinelerra* пригодна для создания и монтажа анимации. Безусловно, *Cinelerra* – не графический редактор и не предоставляет возможности рисовать мультик по кадрам. Все графические материалы надо подготовить в других программах.

Например, вы можете создать анимацию в каком-нибудь пакете 3D-графики и вывести последовательность кадров в виде нескольких сотен (или даже тысяч) TIFF-изображений. Другой вариант – вы покадрово нарисовали движущегося персонажа при помощи планшета, а теперь хотите поместить своего героя на статичный фон или видеоподложку. Либо – вы делаете пластилиновый мультфильм.

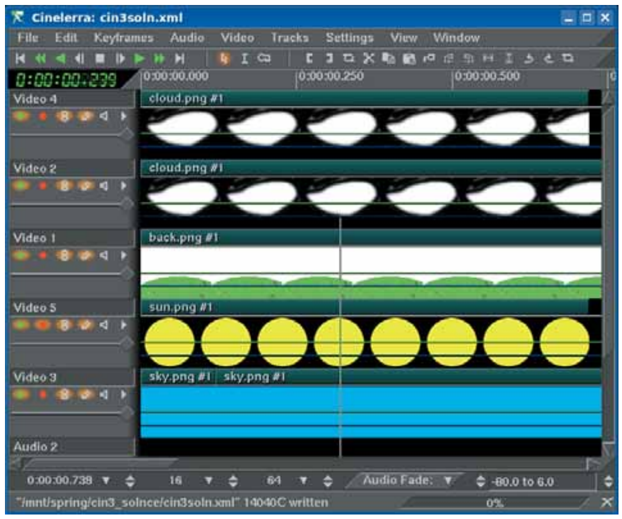
Слоеный пирог

При работе с *Cinelerra* анимацию можно разделить на, скажем так, послойную и покадровую. Что я подразумеваю под послойной анимацией? Допустим, вы хотите сделать сцену – зеленый холм, над ним встает солнце. Вам необязательно покадрово рисовать холм и солнце в разных положениях – низко, выше, еще выше и так далее. Можно поступить проще .

Рисуете холм. Зеленая травка, небо прозрачное. Сохраняете как полноцветный PNG. Отдельно делаете файл с небом – залейте все изображение одним цветом или какой-либо текстурой. После этого в новом файле рисуете на прозрачном фоне солнце и тоже сохраняете в формате PNG. Можете сделать еще и облака, опять-таки каждое на прозрачном фоне. Затем расположите картинки на дорожках в *Cinelerra* следующим образом:

- Сначала, на самой нижней дорожкой, в нижнем слое, небо;
- Затем – солнце;
- После него – холм.;
- Облака можете ставить как «под» холм, там и над ним (если они не будут у вас двигаться вровень с землей).

В окне монтажного стола должно получиться примерно следующее (да простит меня читатель за наивность картинки, но я не художник):

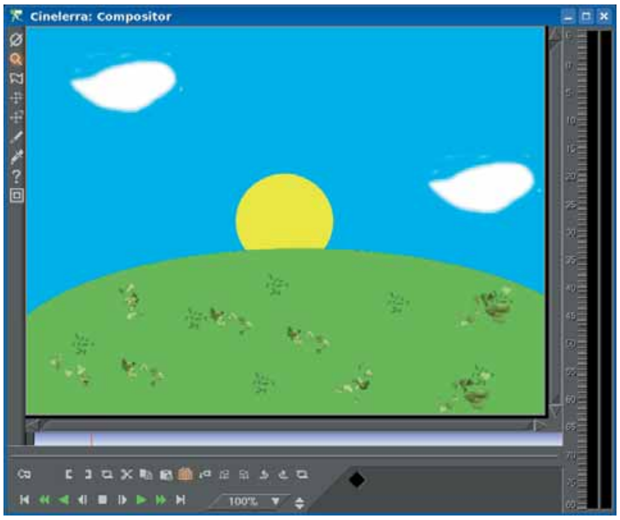


› **Рис. 1.**

Теперь надо поработать в окне Составителя (Composer). Используя проектор (см. LXF93), размещаем по вкусу облака и солнце. Получается вот что:

Месяц назад Мы научились обращаться с камерой и проектором.

аниматора



› **Рис. 2.**

Что до анимации, то она достигается за счет автоматизации. Вы можете автоматизировать проекторы облаков и солнца, тем самым перемещая объекты по экрану. Облака будут плыть по небу, солнце – всходить или заходить по вашему желанию. А на изображение холма и неба можно наложить какой-нибудь эффект, изменяющий цвета, и таким образом делать утро, день и вечер, тоже автоматически.

Иллюзия глубины

Послойное построение сцены позволяет использовать в анимации параллакс. Согласно определению, данному в русской Википедии, параллакс – это «изменение видимого положения объекта относительно удаленного фона в зависимости от положения наблюдателя». Параллакс использовался в старых играх-бродилках для консолей, в нынешних Java-играх для сотовых, да и в обычной мультипликации применяется довольно часто. Допустим, есть некий пейзаж – перечисляю объекты послойно, по мере приближения к зрителю:

1. Небо с облаками.
2. Горы.
3. Роща вдали.
4. Деревья ближе.
5. Кусты на переднем плане.

Мы хотим плавно панорамировать этот пейзаж горизонтально. При смещении камеры, небо может быть статичным либо перемещаться незначительно (относительно кадра), горы будут перемещаться относительно кадра чуть быстрее, роща – еще быстрее, и так далее. Это создает у зрителя иллюзию глубины пространства. Итак, чем глубже находится слой, тем медленнее он движется.

Как сделать параллакс в *Cinelerra*? С послойным составлением кадра мы уже разобрались. Осталось только обратить внимание на некоторые подробности.

Чем ближе слой, тем большего размера он должен быть. Пример: есть два слоя, на одном фоновый пейзаж, на другом – деревья. Пусть панорама длится пять секунд. За эти пять секунд вам надо будет переместить виртуальную камеру слоя с деревьями на большее расстояние, чем виртуальную камеру слоя с пейзажем. Поэтому передний слой должен быть шире, чем фоновый. Чтобы перемещать слои относительно кадра (положения наблюдателя), можно применять такие средства Составителя, как Проектор или Камера. Мне кажется, что логичнее пользоваться последним.

По кадрам

Теперь поговорим о покадровой анимации. Допустим, у вас есть тысяча кадров, каждый в виде PNG- или TGA-файла. Вам нужно как-то поместить их всех на дорожку в *Cinelerra* и монтировать с другими последовательностями кадров или с видео.

»

«Нам помнится, вороне...»

Создать пластилиновый мультфильм, хоть это и трудоемко, вообще-то по плечу каждому человеку, если, конечно, у этого каждого есть цифровой фотоаппарат, чтобы снимать покадрово пластилиновое действо: рисовать умеют не все, а слепить какую ни на есть фигурку с руками-ногами – дело нехитрое. Вот несколько советов, если вы решите этим заняться:

1. В качестве костей скелета используйте обычные канцелярские скрепки. Их можно соединять друг с другом – получатся суставы, и можно делать руки и ноги, подвижные в коленях и локтях.
2. Если вы хотите монтировать фигуры на некой подложке, то фон надо сделать однотонным, чтобы потом его вырезать (см. LXF92). В домашних условиях однотонный фон сделать сложно: плохое освещение, кроме того, будет мешать тень (снимать придется так, чтобы на фон не падали тени от фигур). Неоднородный тон надо будет вырезать в GIMP'е вручную, в каждом кадре-файле, а потом сохранить в полноцветный PNG с прозрачным фоном.
3. Приступая к съемке, надо соотнести количество кадров в секунду в вашем *Cinelerra*-проекте с тем, что вы снимаете. Допустим, в *Cinelerra* у вас задано 25 кадров в секунду. Это значит, что для плавной анимации длительностью в 1 секунду вам надо снять фотоаппаратом 25 кадров. Другие 25 – это пойдет уже вторая секунда анимации. Подход трудоемкий. Если для вас приемлема некоторая дерганность движений, можно снимать по 15, даже 13 кадров для каждой секунды и монтировать их в *Cinelerra* таким образом, чтобы эти 13 кадров отображались 25 раз вместо 13.

» Первое желание – это импортировать кадры как любой другой материал. Но в таком случае в окне Resources на странице Media будет тысяча файлов. Размещать их на дорожке, мягко говоря, не совсем удобно. К счастью, *Cinelerra* может работать с так называемыми toc-файлами. Toc – это сокращение от Table Of Contents, то есть «оглавление». Toc-файл – обычный текст, содержащий список имен файлов. Он также имеет заголовок, в котором указывается частота кадров и их размер (разумеется, все кадры последовательности должны быть одного размера).

Можно создавать toc-файлы вручную, но среди пользователей *Cinelerra* популярна утилита *Seven Gnomes* (http://www.roxton.kiev.ua/sevengnomes.html), которая предоставляет для этих целей удобный графический интерфейс. В *Seven Gnomes* надо в основном (и единственном) окне перейти на вкладку *Index images list* и бросить на открывшуюся страницу папку с кадрами из любого файлового менеджера. После этого начнется создание toc-файла. Здесь же, на странице *Index images list*, задается его имя (можно использовать как просто имя файла, так и полный путь) и частота кадров. Размер кадра определяется программой автоматически. Если не указан полный путь к создаваемому toc-файлу, то файл будет создан в каталоге, где расположены файлы с кадрами.



› **-Рис. 3.**

Seven Gnomes была написана мною пару лет назад в ходе работы над пластилиновым мультиком, а потом, после взятия ее на вооружение сообществом пользователей и разработчиков *Cinelerra*, несколько раз обновлялась. Для сборки программы из исходных текстов нужен только GTK+ 2.x.

И еще кое-что о *Seven Gnomes*. При повторном создании toc-файла для одного и того же каталога, старый toc-файл затирается и не попадает в список файлов (что непременно произошло бы при создании toc-файла вручную). *Seven Gnomes*, как и *Cinelerra*, «понимает» кадры в таких форматах, как TGA, TIFF, PNG и JPEG. Перетаскивать надо не отдельные файлы, а именно каталог.

Итак, тем или иным способом toc-файл получен; что делать дальше? Импортируйте его в проект *Cinelerra* как обычный видеофайл. Toc-файл появится в списке Media окна Resources. Можете обращаться с ним как с любым другим видеофайлом или одиночным кадром – то есть располагать на дорожке, перемещать, усекать и так далее.

Подведем итог. В *Cinelerra* можно использовать и послойную, «программную» анимацию, и покадровую. Можно также сочетать оба подхода. Если вы смотрите мультфильмы, то наверняка заметили, что сцены бывают сделаны несколькими способами:

- На статичном фоне – покадрово анимированные персонажи.
- На фоне с параллаксом – покадрово анимированные персонажи.
- Реже (как более трудоемкая работа) – покадрово анимирован и фон, и персонажи.

Cinelerra подходит для всех трех пунктов.

Отведенное мне место подходит к концу, и напоследок я хочу дать несколько советов. Не забывайте об эффекте Rotate – с его помощью можно вращать объекты на произвольный угол (хотя в документации сказано, что доступны только 4 значения: 0, 90, 180 и 270 градусов).

Для импортированных одиночных кадров можно задать длительность по умолчанию. Делается это в Preferences > Recording > Images > Import images with a duration of <...> seconds. Допустим, вы поставили там 5 секунд и импортировали какой-нибудь PNG-файл. Теперь помещаем его на дорожку – и видим: картинка заняла на дорожке ровно 5 секунд.

Дополнительных эффектов в анимации можно достичь, экспериментируя с ползунком прозрачности на дорожке, а также с режимами наложения (overlay).

При автоматизации Проектора или Камеры, помните, что когда вы включаете режим Generate keyframes while tweaking (нужная для этого кнопка выглядит как ключик), то ключевой кадр в текущем положении курсора в момент включения не создается! Он появляется только при последующем перемещении курсора и передвижении Камеры или Проектора, поэтому автоматизацию надо делать, условно говоря, с конца. Например, есть солнце, и мы хотим переместить его снизу вверх:

- Помещаем его внизу.
- Включаем «ключик». Ключевой кадр в положении солнца «внизу» не создался.
- Смещаем курсор вправо на N секунд, помещаем солнце в верхнюю точку его траектории. Автоматически создается ключевой кадр.
- Смещаем курсор влево (к началу анимации солнца). Помещаем солнце вниз. Вот теперь-то и создается первый ключевой кадр, с указанием Проектору, что солнце надо выводить внизу кадра.

Если вы снимите собственный мультфильм и смонтируете его в *Cinelerra*, не забудьте сообщить нам об этом (letters@linuxformat.ru) – видеотека нашего главного редактора вот-вот закончится. LXF

Во славу альфа-канала

Как вы могли заметить, послойная анимация возможна благодаря поддержке прозрачности в формате PNG. Напомню, что PNG позволяет хранить изображения в двух цветовых режимах. Первый – он используется в основном в Сети – это режим палитры, а второй – с 24-битным цветом, что соответствует стандартному BMP. Прозрачность PNG вообще можно использовать для разных нужд. Например, взять изображение телевизора, прорезать в экране дырку (то есть сделать область, занимаемую экраном, прозрачной), и потом в этот телевизор что-то «вставить» – а на деле просто разместить дорожку с телевизором над видеоподложкой.

Через месяц От мультфильмов к документалистике: мы займемся монтажом видео.

OpenVPN за 10 минут

Технология VPN во многом напоминает перевозку денег в бронированной инкассаторской машине по улицам многолюдного города. Но какую модель следует выбрать, чтобы ваши ценности не достались злоумышленникам? Спросите у **Валентина Яценко**.

VPN – *Virtual Private Network*, Виртуальная Частная Сеть – технология, позволяющая соединять между собой локальные сети и отдельные компьютеры, находящиеся на расстоянии тысяч километров друг от друга; подключаться к локальной сети через Интернет, используя шифрованный канал связи, организовывать безопасный туннель для передачи важных данных... Возможности применения VPN безграничны! Вам необходимо соединить офисы в разных концах страны единой сетью, не прокладывая километры кабелей? Вы хотите подключаться к домашней локальной сети из Интернет-кафе? Требуется передать важные данные на большое расстояние без риска утечки информации? Может быть, большинство данных, которые вы передаете через Интернет, составляют тайну, и их необходимо обезопасить? Если вы хотя бы на один вопрос ответили утвердительно, VPN – это именно то, что вам нужно!

Шаг 1: Устанавливаем

Пакеты с *OpenVPN* доступны для большинства распространенных дистрибутивов, в том числе для RedHat/Fedora, Mandriva, OpenSUSE, *Slackware*, Debian и Ubuntu. Если *OpenVPN* не включен в ваш любимый дистрибутив, попробуйте поискать пакет на **rpmfind.net** (для систем, использующих RPM) или **linuxpackages.net** (Slackware и производные) Наконец, программу можно загрузить в исходных текстах с официального сайта (**openvpn.net**).

Почему OpenVPN?

Существует множество реализаций VPN, самые популярные из них – PPTP, *FreeS/WAN* и *OpenVPN*. Почему же в этой статье мы рассмотрим именно последний вариант? PPTP навевает мысли о Microsoft, *FreeS/WAN* давно не обновлялся – на его официальном сайте сказано, что последняя версия датируется 22 апреля 2004 года, более того, дальнейших обновлений *FreeS/WAN* не будет, а это, согласитесь, недопустимо для проекта, одной из целей которого является обеспечение безопасности... *OpenVPN* же активно развивается и регулярно обновляется, использует стойкие криптографические алгоритмы и предлагает множество способов идентификации, в том числе комбинированных, что немаловажно для безопасности. Кроме того, *OpenVPN* работает на большинстве распространенных платформ – Linux, Windows 2000/XP, OpenBSD, FreeBSD, NetBSD, Mac OS X, Solaris, благодаря чему его можно использовать в гетерогенных сетях. Все это, безусловно, делает *OpenVPN* лучшим выбором для организации VPN.

Для установки *OpenVPN* также необходимо установить *OpenSSL* (**openssl.org**) и *liblzo* (**http://www.oberhumer.com/opensource/lzo**), которые есть во многих дистрибутивах. Для работы *OpenVPN* нужен модуль ядра *tun*. Он присутствует в большинстве систем, но если команда modprobe tun завершается с ошибкой «Can't locale module tun», вам придется пересобрать ядро – о том, как это сделать, подробно рассказано в LXF89. Необходимо включить опцию CONFIG_TUN (Device Drivers –> Network device support –> Universal TUN/TAP device driver support).

Если вы собираетесь использовать *OpenVPN* постоянно, пропишите модуль *tun* в **/etc/modules** для автоматической инициализации при загрузке либо включите его непосредственно в ядро. В противном случае перед запуском *OpenVPN* вам нужно будет загружать модуль *tun* самостоятельно (командой modprobe tun). Если вы планируете предоставлять клиентам доступ в Интернет через *VPN*, необходимо настроить NAT (Network Address Translation). Скорее всего, это уже сделано за вас разработчиками дистрибутива, но на случай возникновения проблем имейте в виду, что нам потребуется опция CONFIG_IP_NF_TARGET_MASQUERADE (Networking –> Networking options –> Network packet filtering –> IP tables support –> Full NAT). Для работы с NAT используется утилита *iptables* (см. LXF87/88), которая, скорее всего, уже установлена в вашей системе. Проверьте это и, в случае необходимости, добавьте ее через менеджер пакетов вашего дистрибутива.

LINUX FORMAT на DVD
» Примеры конфигурационных файлов

»

Шаг 2: Создаем ключи

Итак, *OpenVPN* установлен. Что дальше? В первую очередь необходимо создать ключи и сертификаты для сервера и клиентов. Скрипты, облегчающие эту задачу, находятся в каталоге **easy-rsa** (*см. врезку «Где же easy-rsa?» внизу*).

В первую очередь, отредактируйте расположенный в нем файл vars. Установите значения переменных:

- **KEY_COUNTRY** – равным двухсимвольному коду вашей страны (для России это RU).
- **KEY_PROVINCE** – названию вашего региона (например, Moscow region или MSK).
- **KEY_CITY**, соответственно – ваш город (скажем, Moscow).
- **KEY_ORG** – организация (OpenVPN Solutions подойдет).
- **KEY_EMAIL** – ваш e-mail адрес (me@my.e-ma.il).

Все эти данные необходимы для генерации сертификатов и ключей в соответствии со стандартом, поэтому оставлять какие-то переменные пустыми нельзя.

Также можно изменить параметр KEY_SIZE, отвечающий за размер ключей (максимальный размер ключа, поддерживаемый *OpenVPN* – 2048 бит).

Далее необходимо инициализировать переменные, используемые при создании публичного ключа командой

```
. ./vars
```

и, на всякий случай, удалить существующие ключи командой

```
./clean-all
```

После этого следует сгенерировать корневой сертификат (см. LXF93):

```
./build-ca
```

В процессе выполнения последнего скрипта вам будет задано несколько вопросов. Нужно ввести только Common Name для сертификата (например, RootCA), все остальные параметры уже заданы в качестве переменных окружения скриптом vars.

Скорая помощь

Не обязательно создавать корневой сертификат вручную. Подробности см. на стр. 94

Где искать easy-rsa?

Каталог **easy-rsa** может находиться в **/usr/share/doc/openvpn-2.0**, **/usr/share/doc/openvpn**, **/usr/share/doc/openvpn/examples** или в **/usr/share/doc/packages/openvpn** – в зависимости от дистрибутива. Его также можно найти в архиве с исходными текстами.

После создания корневого сертификата необходимо сгенерировать для сервера секретный ключ. В этом вам поможет скрипт *build-key-server*. Как и в предыдущем случае, для большинства вопросов можно оставить ответы по умолчанию. На вопрос о Common Name введите “server”, на два последних вопроса (подписать ли сертификат и – каламбур, но перевод именно такой – сертифицировать ли его) ответьте утвердительно; когда вас спросят о пароле (“A challenge password”) – введите пароль для ключа сервера (запоминать его не нужно). Наберите

```
./build-key-server server
```

Далее, создайте ключи для необходимого количества клиентов. Эта процедура аналогична созданию ключа для сервера, за исключением того, что для клиентов ключи создаются скриптом build-key, а не build-key-server. У каждого ключа, конечно же, должно быть указано свое значение в поле Common Name:

```
./build-key clientname
```

Если вы хотите защитить паролем ключи клиентов, воспользуйтесь вместо *build-key* скриптом *build-key-pass*.

И наконец, в довершение ко всем сертификатам и ключам необходимо создать файл параметров Диффи [Diffie] и Хеллмэна [Hellman]. Алгоритм Диффи и Хеллмэна позволяет обмениваться секретными ключами по небезопасным каналам. Для их генерации запустите скрипт *build-dh* без параметров:

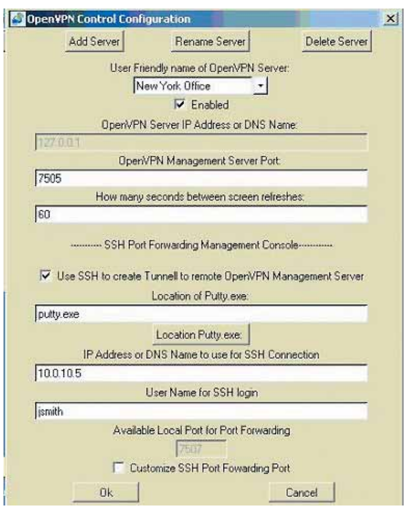
```
./build-dh
```

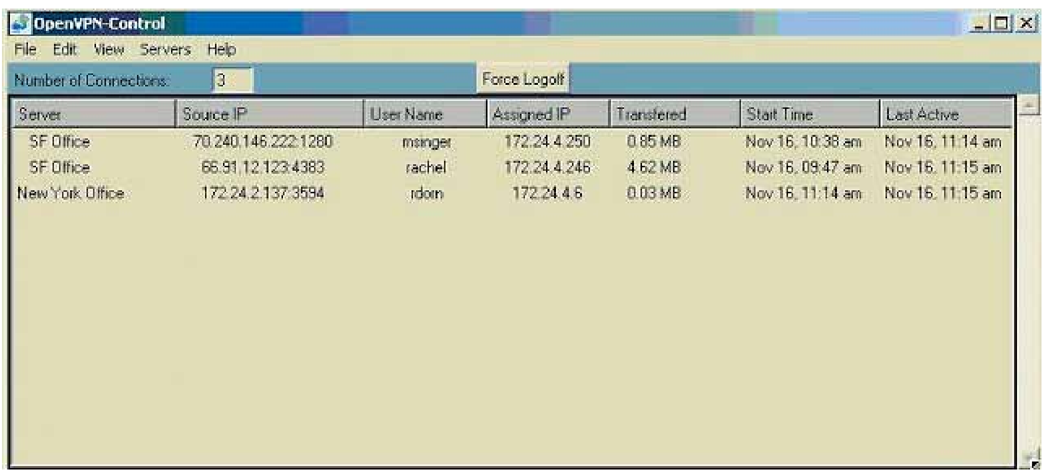
Этот процесс занимает определенное время, поэтому владельцам слабых компьютеров стоит попить чаю, вместо того чтобы просто сидеть и ждать окончания работы скрипта.

Если вам уже надоело возиться с ключами, сертификатами и прочим – не падайте духом: осталось только скопировать нужные файлы на клиентские компьютеры. Клиентам необходимы следующие файлы:

- **ca.crt** (Root CA certificate) необходим серверу и всем клиентам, он не является секретным.
- **clientname.crt** (сертификат клиента clientname) необходим только клиенту clientname, он не является секретным.
- **clientname.key** (ключ клиента clientname) необходим только клиенту clientname, это секретный ключ.

Эти файлы можно найти в подкаталоге **keys** каталога **easy-rsa**.





Шаг 3: Настраиваем

Итак, у нас есть все необходимые для работы *OpenVPN* файлы – все, кроме конфигурационных, созданием которых мы и займемся. Начнем с сервера. Для создания новых конфигурационных файлов удобно использовать примеры из поставки *OpenVPN*, они находятся в подкаталоге **sample-config-files** каталога *OpenVPN* (**../sample-config-files**, если вы находитесь в каталоге **easy-rsa**). В принципе, можно использовать стандартный конфигурационный файл для сервера, изменив некоторые параметры. Вот список этих параметров (некоторые из них, например, port и protocol, изменять не обязательно):

- **ca** – полный путь до корневого сертификата, который мы создали скриптом *build-ca*.
- **cert** – полный путь до сертификата сервера, который мы создали его скриптом *build-key-server*.
- **key** – путь до ключа сервера, созданного все тем же скриптом *build-key-server*.
- **dh** – путь до файла параметров Диффи и Хеллмэна, созданного скриптом *build-dh*.
- **server** – режим запуска *OpenVPN*. В строке с режимом запуска для сервера должна быть указана подсеть и маска подсети (пример – server 10.8.0.0 255.255.255.0). Подсеть, указанная по умолчанию – 10.8.0.0/24. Если в вашей сети она уже используется, задайте другую локальную подсеть.
- **port** – порт, на котором будет запущен сервер. Стандартный порт *OpenVPN* – 1194, но вы можете изменить его по своему желанию.
- **protocol** – протокол. По умолчанию *OpenVPN* использует экономичный UDP, но вы можете предпочесть надежный TCP.

Если вы не сторонник конфигураций по умолчанию, отредактируйте файл в соответствии со своими предпочтениями. Если же вы не хотите тратить на это время, то можете использовать мой вариант, составленный специально для этой статьи. Вы найдете его на прилагаемом к журналу диске.

Для создания конфигурационного файла клиента также можно использовать пример, идущий в поставке с *OpenVPN*. В нем надо изменить пути к необходимым файлам на свои, указать адрес и порт сервера в директиве remote и, кроме того, удостоверится, что выбранные настройки совпадают с настройками сервера (протокол, сжатие и т.п.). Пример конфигурационного файла клиента можно также взять с диска LXF. Тех же, кто хочет предоставлять *VPN*-клиентам доступ в Интернет, ждет еще один шаг: включение NAT (Network Address Translation, Трансляция сетевых адресов) и настройка маршрутизации. Для начала необходимо написать соответствующее правило *IPTables*:

```
iptables -t nat -A POSTROUTING -s маска_подсети -o внешний_
интерфейс -j MASQUERADE
```

например:

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o ppp0 -j MASQUERADE
```

Его необходимо сохранить, добавив в файл с правилами iptables или в скрипты загрузки. Затем необходимо включить маршрутизацию, записав в **/etc/sysctl.conf** строку “net.ipv4.ip_forward = 1” и выполнив команду

```
sysctl -w net.ipv4.ip_forward=1
```

Шаг 4: Проверка связи

Если вы дошли до этого места, спешу вас обрадовать – остался всего один, очень простой шаг. Надо запустить *OpenVPN* на сервере, а потом и на всех клиентах. Для сервера: перейдите в каталог с ключами (keys) и выполните:

```
openvpn --config ../sample-config-files/server.conf
```

После чего вы должны увидеть подобные строки:

```
Thu Aug 31 12:04:20 2006 OpenVPN 2.0.7 i486-pc-linux-gnu [SSL]
[LZO] [EPOLL] built on Jul 22 2006
Thu Aug 31 12:04:20 2006 Diffie-Hellman initialized with 1024 bit key
Thu Aug 31 12:04:20 2006 TLS-Auth MTU parms [ L:1542 D:138 EF:38
EB:0 ET:0 EL:0 ]
Thu Aug 31 12:04:20 2006 TUN/TAP device tun0 opened
<Skipped>
Thu Aug 31 12:04:20 2006 Initialization Sequence Completed
```

На клиенте (клиентах) выполните команду

```
openvpn --config client.conf
```

после чего проверьте наличие tun-интерфейса командой ifconfig:

```
ifconfig tun0
tun0 Link encap:UNSPEC HWaddr 00:00:00:00:00:00
inet addr:85.140.161.153 P-t-P:10.8.0.2 Mask:255.255.255.0
UP POINTOPOINT RUNNING MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:10
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
```

Обратите внимание на статус (UP): он означает, что интерфейс находится в активном состоянии. Если в вашей системе имеется более одного tun-интерфейса (это можно проверить командой ifconfig без параметров), замените tun0 на tun1, tun2 и т.п.

Если интерфейс сконфигурирован правильно, проверьте доступность других хостов. Когда путь к файлу соответствий «клиент-IP» (опция ifconfig-pool-persist в конфигурационном файле сервера) не указан, их адреса назначаются сервером *OpenVPN* случайным образом. Наберите:

```
ping 10.8.0.2
```

Вместо 10.8.0.2 подставьте IP-адрес, выданный командой ifconfig на другой машине. В случае, если интерфейс сконфигурирован правильно, но ответа не приходит, необходимо выяснить, разрешают ли правила межсетевого экрана ICMP-трафик.

Приходящие ответы свидетельствуют о том, что *VPN*-соединение установлено и работает. LXF





Ответы

Есть вопрос по OpenSource? Пишите нам по адресу: answers@linuxformat.ru

1 Решительный шаг

Хоть я и новичок в Linux, но все же решился полностью убрать Windows XP со своего ноутбука. У меня двойная загрузка XP и Ubuntu; не могли бы вы подсказать мне, как убрать Windows и оставить только Ubuntu? Как передать Linux место, прежде занятое Windows XP? А может быть, мне, как новичку, проще отформатировать весь диск и переустановить Linux?

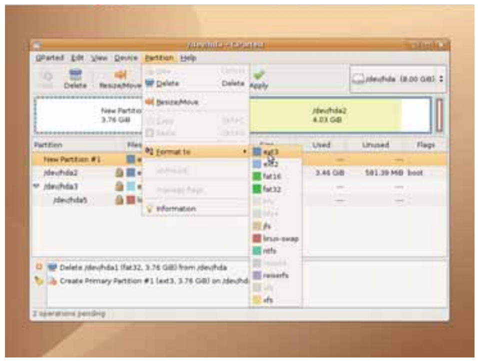
Pub Bloke, из форумов LXF

Начнем с вашего последнего вопроса. Конечно, переустановить Ubuntu и отдать ему во владение весь диск проще, но тогда вы потеряете все свои настройки и данные. Удаление Windows-раздела и перераспределение освободившегося места под Linux оставит текущую установку Ubuntu нетронутой. Кроме того, проделав эту работу, вы ближе познакомитесь с Linux.

Уничтожить Windows нетрудно. Сначала нужно удалить Windows-раздел (обычно **hda1**) с помощью *Gnome Partition Editor*, который вы найдете в меню System > Administration (Система > Администрирование). Если его там нет, добавьте *GParted* через менеджер пакетов *Synaptic*.

Найти Windows-раздел тоже легко, обычно он содержит файловую систему NTFS (в крайнем случае, FAT), а Linux такими файловыми системами не пользуется. Удалив раздел, щелкните мышью в освободившейся неразмеченной области и нажмите кнопку New (Создать) для создания нового Linux-раздела с файловой системой ext3 (сгодятся установки по умолчанию).

Пока новый раздел выделен в списке, перейдите в меню и выберите Partition > Format To > Ext3 (Раздел > Форматировать в > Ext3) (*см. экранный снимок справа*). Нажмите Apply (Применить), чтобы изменения вступили в силу.



› ***GParted* удалит Windows-раздел и заменит его Linux-разделом.**

Следующий шаг – удаление пункта Windows из загрузочного меню. Откройте терминал и введите

```
sudo -i
gedit /boot/grub/menu.lst
```

чтобы открыть загрузочное меню в редакторе. Ближе к концу этого файла вы найдете строку, начинающуюся с 'title Windows'. Удалите все вплоть до следующей пустой строки и сохраните файл. Windows в загрузочном меню больше не появится.

А вот освоить только что освободившееся пространство уже сложнее. Увеличить Linux-раздел можно, только отодвинув его окончание, а освобожденное пространство располагается перед началом Linux-раздела, так как Windows-раздел был первым на диске. К счастью, Linux может использовать несколько разделов – в данном случае вы можете занять освободившееся место под свою домашнюю директорию (преимущество такого подхода состоит в том, что при переустановке системы или переходе на новый дистрибутив вы сохраните все свои персональные данные, поэтому они находятся на отдельном разделе). Сообщить системе о том, что данный раздел нужно занять под домашнюю директорию, можно, добавив строку в файл **/etc/fstab** (это таблица файловых систем). В терминале, которым вы только что пользовались, введите

```
gedit /etc/fstab
```

Добавьте следующую строку и сохраните файл:

```
/dev/hda1  /home  ext3  defaults  0 0
```

До перезагрузки, которая активирует новый home-раздел, скопируйте туда существующие файлы. Не покидая терминал, введите

```
mkdir /mnt/tmp
mount /dev/hda1 /mnt/tmp
mv /home/* /mnt/tmp/
reboot
```

Эти команды временно смонтируют новый раздел в **/mnt/tmp**, переместят в него ваш домашний каталог и перезагрузят компьютер, после чего изменения станут необратимыми. Теперь в загрузочном меню не останется упоминания о Windows, а после загрузки Ubuntu то место, что прежде занимала Windows, мож- »

Наши эксперты

» Мы найдем эксперта для ответа на любой вопрос, от инсталляции и установки модемов до администрирования сети – лишь черкните письмо, а об остальном мы позаботимся.



Нейл Ботвик
Поруководив небольшим Интернет-провайдером и поизготавливав сопроводительные диски для LXF, Нейл считает себя «Linux-мастером на все руки».



Майк Сондерс
Майк стоял у истоков Linux Answers в LXF. Его специализация – программирование, оконные менеджеры, init-скрипты и SNES.



Стефан Лукас
Пол – местный супер-программист, но справится также и с вашими проблемами с Сетью и базами данных.



Ник Вейч
Когда он не редактирует что-нибудь большим красным карандашом, Ник занимается графическим и трехмерным ПО для Linux. Готов ответить даже на самый простой вопрос!



Валентин Синицын
В редкие свободные минуты главный редактор нашего журнала обычно запускает mcedit, чтобы отшлифовать какое-нибудь открытое приложение. Его любимая тема – настольный Linux.

КУДА ПОСЫЛАТЬ ВОПРОСЫ:
Пише на м по адресу: answers@linuxformat.ru или спрашивайте на форуме: www.linuxforum.ru

» но будет использовать для хранения Ваших собственных файлов. НБ

2 Прокси для деток

В Web-прокси *Squid* (и *SquidGuard*) работает на моем компютере с Fedora Core 6 с тех пор, когда я прочел о нем в самом первом учебнике Hardcore Linux, в LXF75. Я настроил *SquidGuard* так, чтобы защитить моих детей от нежелательного контента. Это означает, что на их (Windows XP) компьютере я установил маршрут доступа в Интернет через мой прокси-сервер (192.168.100.100:8080), и все бы прекрасно.

Но меня беспокоит, что мой старший очень быстро взрослеет, и недалек тот день, когда он догадается снять флажок Use Proxy Server (Использовать прокси-сервер) и, настроив прямое подключение к Интернету, получит ничем не ограниченный доступ. Можно ли пропускать весь трафик через мой (постоянно включенный) FC6-компьютер (вероятно, пробросом портов на маршрутизаторе, пароль к которому знаю только я), чтобы весь web-трафик проходил через прокси-сервер, а при «прямом» подключении доступа в Интернет не было? Если можно, то как? Я пробовал перенаправить порты 80 и 8080 на IP моего компьютера, но это не помогло.

Марк [Mark], из форумов LXF

О Насколько я понимаю, под словом «Интернет» вы подразумеваете World Wide Web, то, с чем обычно имеет дело *Squid*. Однако вы можете направить весь Интернет-трафик через свою FC6-машину, а следовательно, через *SquidGuard*, в три шага.

Во-первых (конкретное воплощение зависит от вашего маршрутизатора), вы должны настроить маршрутизатор так, чтобы он позволял подключаться к Интернету только Вашей FC6-машине. Сделанный Вами проброс портов влияет только на входящие подключения, поэтому ликвидируйте его.

Во-вторых, необходимо назначить FC6 шлюзом по умолчанию, чтобы весь Интернет-трафик (а не только web-трафик) проходил через него. Отредактируйте **/etc/sysctl.conf** от имени суперпользователя, изменив окончание строки

```
net.ipv4.ip_forward = 0
```

с 0 на 1. Потом наберите

```
service network restart
```

Теперь нужно перенастроить детский компьютер на использование IP-адреса вашего компьютера в качестве адреса сетевого шлюза. Так как вы запретили им бесконтрольный доступ с помощью маршрутизатора, теперь это единственный путь подключения к Сети.

Остается проблема удаления Вашими детьми всяких прокси-настроек, поэтому мы воспользуемся функцией *Squid*, называемой «прозрачный прокси». Эта функция принудительно направляет все web-запросы, проходящие через машину – а вы уже предопределили их направление на предыдущих шагах – на прокси *Squid*, следовательно, на *SquidGuard*. Откройте в редакторе файл настройки *Squid* (обычно **/etc/Squid/Squid.conf**) и найдите строку (строки), начинающиеся с 'http_port'. В Вашем файле это, скорее всего, выглядит так: **http_port 8080**. Измените это на

```
http_port 80 transparent
```

80 – это стандартный HTTP-порт. Параметр **transparent** предписывает *Squid* вмешиваться и перехватывать все запросы, независимо от того, настроен браузер на работу с прокси или нет. вам нужно или удалить старую настройку прокси из браузеров, или добавить строку обработки запросов для старого порта 8080.

```
http_port 8080 transparent
```

Существует другой метод сделать то же самое. Можно оставить http_port на 8080, а запросы, которые вы хотите пропускать через прокси, перенаправить с порта 80 на 8080 при помощи правила *Iptables*. Это сложнее, но удобнее, так как позволяет некоторым машинам полностью миновать прокси. Подробнее об этом можно узнать на web-сайте *Squid*: www.Squid-cache.org.

Вы можете также использовать *Iptables* или графическую оболочку вроде *Firestarter* для блокировки исходящего трафика на любые порты, кроме стандартных (HTTP, HTTPS, POP3, SMTP и FTP). Это предотвратит использование Вашими детьми удаленного прокси, работающего через другой порт. Возможно, вы решите сделать это на маршрутизаторе, но если вы установите запрет на своей FC6-машине, то блокировка нежелательного доступа для детей не ограничит вашей собственной свободы. НБ



› **Прозрачный прокси отвлекает трафик на себя, но необходим брандмауэр, чтобы перекрыть обходные пути.**

Краткая справка об…

X-пробросе

Легкий путь к удаленной работе.

X, основа всех графических интерфейсов Linux – сетевой протокол. Чаще всего ваш дисплей подключается к X-серверу на локальной машине, но это совершенно не обязательно. Для доступа к рабочему столу удаленного компьютера вы можете пользоваться VNC или NX, но этот метод имеет один недостаток: необходимо воспроизводить весь рабочий стол другого компьютере. А еще нужно, чтобы на удаленном компьютере реально имелся рабочий стол. Если этот компьютер – автономный сервер, запуск рабочего стола может оказаться растратой ресурсов.

Вместо запуска полномасштабного рабочего стола можно запустить лишь то приложение, которое вам необходимо, через SSH-соединение, безопасное даже при работе через общедоступные сети. Наберите в своем терминале следующее:

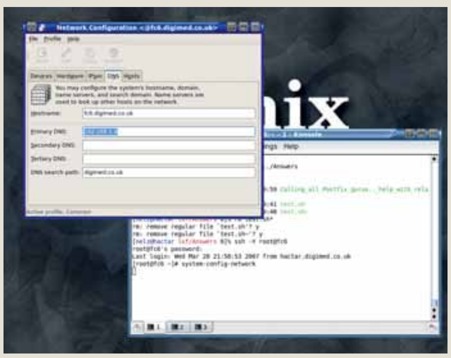
```
ssh -X user@remote.computer
someprogram
```

Если **someprogram** – графическое приложение, то его окно откроется на вашем рабочем столе. Правда, кое о чем нужно помнить. Во-первых, у вас должен быть SSH-доступ к другому компьютеру через пароль или по ключу. Кроме того, на SSH-сервере другого компьютера должен быть настроен проброс X (X forwarding). Он обычно настраивается по умолчанию, но на всякий случай проверьте, что **/etc/ssh/sshd_config** содержит

```
X11Forwarding yes
```

У ssh есть и другие полезные функции. Например, можно использовать **-Y** вместо **-X**. Этот параметр доступен, если SSH-сервер настроен на Trusted X11 Forwarding, и существенно ускоряет процесс передачи данных. Другой параметр, способный ускорить подключение по медленным линиям – это **-C**, вызывающий сжатие всех данных. Хотя X для медленных подключений все же не годится – лучше пользоваться *NX* или *TightVNC*.

Mac OS тоже использует *X* на своих дисплеях, поэтому вы можете видеть окна Mac-программ на Linux-мониторах, и наоборот.



› **Запуск программы сетевой настройки Fedora Core на другом компьютере.**

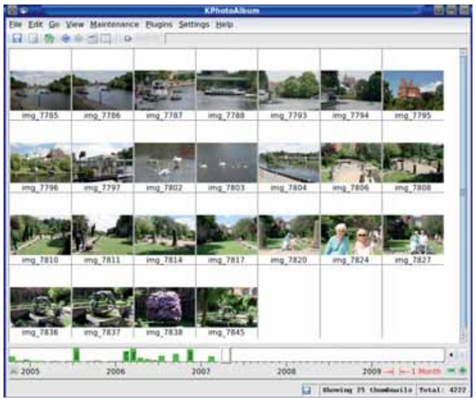
3 Упорядочить хаос

В Отбилась от рук моя фотоколлекция на несколько гигабайт. Я хотел бы упорядочить ее, чтобы сделать резервную копию. Не подскажете ли программу, которая может переименовать файл с учетом данных о времени создания (EXIF) и установить время изменения файла равным времени создания? Моя недавняя попытка сделать резервную копию перед очисткой компьютера привела к тому, что даты создания всех файлов оказались замещены датой записи DVD.

А еще я умудрился рассеять по всей коллекции несколько дубликатов изображений (сам не знаю как), все под разными именами. Нельзя ли это все рассортировать (может быть, тоже на основании EXIF) без пересмотра нескольких тысяч фотографий?

Если это имеет значение, то у меня Fedora Core 6 64-bit, командная строка не пугает.

NiceBloke, из форумов LXF »



› **Менеджеры цифровых фотографий тоже умеют обращаться с EXIF-данными файлов, но консоль эффективнее.**

» **О** С данными EXIF работают несколько программ. Я предпочитаю *ExifTool* (www.sno.phy.queensu.ca/~phil/exiftool). *ExifTool* читает и обрабатывает почти любые данные EXIF, включая извлечение EXIF-тэгов Date/Time Original (первоначальной даты/времени) или Create Data (времени создания). Эту информацию можно использовать для переименования файлов или изменения их временных отметок. Например:

```
find -name '*.jpg' | while read PIC; do
DATE=$(exiftool -p '$DateTimeOriginal' $PIC |  sed 's/[:
]//g')
touch -t $(echo $DATE | sed 's/\(..$\)/\.\1/') $PIC
mv -i $PIC $(dirname $PIC)/$DATE.jpg
done
```

Первая строка находит все файлы ***.jpg** в текущем и дочерних каталогах. Следующая стока извлекает тэг Date/Time Original каждого файла (может понадобиться тэг Create Data, смотря какая камера) и удаляет все пробелы и двоеточия. Затем команда touch устанавливает временную отметку согласно полученным данным, – устрашающее регулярное выражение sed всего лишь вставляет точку перед двумя последними символами (touch ожидает, что секунды будут отделены от прочих данных временной строки именно так). Заключительная строка переименовывает файл, используя параметр -i с mv на тот случай, если временные отметки двух файлов одинаковы. Это предотвращает перезапись файлов.

Подобное умеет проделывать большинство программ обработки цифровых фото, обходящихся без командной строки – *DigiKam*, *KPhotoAlbum*, *F-Spot* и *GThumb* все имеют средства обращения с файлами, основанные на использовании EXIF-данных.

Недостаток этих программ – они работают только в текущей директории, а вышеприведенный скрипт обрабатывает JPEG-файлы во всех вложенных каталогах. Если же у вас несколько гигабайт фотографий в единственной директории, то такую коллекцию вряд ли приведет в порядок одно только переименование файлов!

Решением проблемы дубликатов может стать программа *fdupes* (http://netdial.caribe.net/~adrian2/fdupes.html, доступная в виде RPM для FC6). Программа сравнивает содержимое файлов, поэтому способна находить дубликаты даже под разными именами и временными метками.

```
fdupes --recurse ~/photos
```

выведет все дубликаты в каталоге **photos**. Можно удалить лишние дубли:

```
fdupes --recurse --omitfirst --sameline ~/photos | xargs
rm
```

Соблюдайте осторожность при любых действиях, связанных с автоматическим удалением файлов. Сначала запустите программу без удаления и оцените результат ее работы. **ПХ**

4 ISO: разочарование

В Прочитав LXF91, я решил попробовать систему *Jigdo*, чтобы сделать ISO для Fedora. Можно было бы и поподробнее инструкции в журнале привести. Например, как пользоваться командой *mkiso* и что делать, если DVD смонтирован с поехес.

Нужно ли копировать файловую систему поверх локальной файловой системы, и нужно ли перемещать файл *Jigdo*? Не знаю, делал ли так кто-нибудь. Так или иначе, я получил следующие ошибки:

'sudo *./mkiso*
Creating FC-6-i386-livecd-1.iso
general:	Image file: /home/user//FC-6-i386-livecd-1.iso
general:	Jigdo:	/home/user//FC-6-i386-livecd-1.iso.Jigdo
general:	Template:	Jigdo/FC-6-i386-livecd-1.template
Skipping object `../..//.mozilla/*Firefox*/i4faho56.default/lock' (No such file or directory)
Found 0 of the 5 files required by the template
Will not create image or temporary file - try again with different input files
general:	[exit(1)]
ISO image written to /home/user//FC-6-i386-livecd-1.iso
Verifying MD5 checksums...
md5sum: FC-6-i386-livecd-1.iso: No such file or directory
FC-6-i386-livecd-1.iso: FAILED open or read
md5sum: WARNING: 1 of 1 listed file could not be read
Verification failed, or you do not have the »

Часто задаваемые вопросы...

Что такое LDAP?

Это не просто телефонный справочник!

» Опять какое-то сокращение? Ну и что оно означает?
Оно означает Lightweight Directory Access Protocol (Облегченный протокол доступа к каталогам).

» Как обычно, это мне ни о чем не говорит. Поясните!
LDAP – это протокол для обслуживания и изменения информации в каталогах через сетевое подключение.

» Типа *Yellow Pages* в Интернет?
Вроде того. Каталог LDAP может быть любым набором логически и иерархически организованных данных. Телефонный справочник – подходящий пример: данные там состоят из имен, адресов и телефонных номеров, упорядоченных по алфавиту. Или по категориям, как в *Yellow Pages*.

» Так это телефонная книга онлайн?
Это может быть и телефонной книгой, а может содержать любой другой набор упорядоченных данных; например, число компьютеров в сети, товары и цены на них, каталоги документов… словом, все, что поддается структурной организации. Это даже не обязательно онлайн-структура. Большинство LDAP-серверов работает в локальных сетях или на отдельных машинах. Вряд ли ваша компания захочет распространять сведения о себестоимости продаваемой продукции через Интернет.

» Похоже, что каталог LDAP – один из видов базы данных?
Сходство есть, но неполное. LDAP не предназначен для работы со сложными наборами данных и запросами, отсюда и термин «Облегченный». Продолжая тему, *PostgreSQL* или *Oracle* тяжеловаты для хранения списка офисных сотрудников и их добавочных телефонных номеров. LDAP для этого вполне достаточно.

» Полагаю, его и настроить проще?
Верно полагаете. Особенно после некоторой тренировки. LDAP не база данных SQL, поэтому кое-что придется изучить, но документация подробна и снабжена множеством шаблонных схем.

» Погодите! Что еще за схемы?
Схема определяет строение каталога, атрибуты, которые может иметь каждая запись. Похоже на описание таблицы в SQL.

» Ну, создал я каталог LDAP, и как теперь им пользоваться? Наверное, нужно специальное ПО?
Никакого. Многие популярные программы способны использовать LDAP-каталоги как источник информации. Например, некоторые почтовые клиенты используют LDAP в качестве адресной книги.

» А как же встроенные адресные книги, которыми обычно пользуются почтовые программы?
Они и есть встроенные: работают лишь с одним почтовиком. А можно иметь централизованный источник информации, в котором почтовые программы будут искать адреса, через *Samba* можно будет проводить авторизацию пользователей… словом, это единый источник сведений, который можно использовать так, как вам нравится (или – как велит сисадмин).

» md5sum program installed.
In the latter case, you probably have nothing to worry about.'

Подскажите, пожалуйста, что делать с непонятной ошибкой md5sum, и как обмануть проверку контрольной суммы, чтобы, наконец, создать ISO.

Тодд [Todd]

Во-первых, незачем запускать *mkiso* от имени суперпользователя, ведь единственное место, к которому ей нужен доступ для записи – это ваш домашний каталог. Во-вторых, программу следует запускать из того места, где вы планируете создать образы дисков, или приводить директорию назначения как аргумент скрипта.

Если вы перейдете в каталог Fedora на DVD и запускаете *mkiso*, то программа все же сработает и запишет ISO-образ(ы) в Вашу домашнюю директорию, но не сможет создать свои кэш-файлы. В данном случае это не так уж важно, так как вы записываете единственный образ, но в случае записи нескольких образов потеря времени будет ощутимой.

Если DVD смонтирован с **--noexec**, то скрипт все же будет работать с *sh*, так как *sh* работает с жесткого диска, а скрипт – всего лишь файл данных, обрабатываемый ею:

```
sh /path/to/dvd/Distros/Fedora/mkiso
```

Ключ к решению Вашей проблемы находится в строке:

'Found 0 of the 5 files required by the template'

Почему-то *Jigdo* не смогла найти ни один из нужных ей файлов. Предполагаю, что к этому мог привести запуск *mkiso* от имени суперпользователя, в результате чего скрипт не нашел место, откуда ему следовало запускаться, и те файлы, с которыми он должен был работать (путь поиска относителен месту запуска скрипта).

Можно было бы предположить, что ваш DVD читается с ошибками – но вряд ли это испортит все пять файлов. Такой диск не работал бы совсем.

Обход проверки контрольной суммы MD5? вы предлагаете исправить проблему низкого давления масла в автомобиле отключением индикаторной лампы! Как видно из сообщения системы, проверка контрольной суммы ISO-файла MD5 не удалась только потому, что проверять было нечего.

Кстати, запуск *mkiso* с параметром *-h* выводит подробные сведения о работе программы. **MC**



› **Перед обращением к модему, убедитесь, что удалены все следы широкополосного подключения.**

5 Проблема Firefox

В Мой *Firefox* никак не может «увидеть» модемное подключение, которое я с таким трудом настроил. Я совершенно уверен в том, что подключение рабочее, так как запуск *pon* из командной строки заставляет модем звонить, а *poff* кладет трубку. А вот запуск *Firefox* с рабочего стола превращается в проблему. Широкополосное подключение Ethernet работает отлично, но его удаление и установка модемного подключения «по умолчанию» вызывает появление экрана 'server not found'. Модем Rockwell IQ148, я пользуюсь Ubuntu Dapper 6.06. Я пытаюсь настроить компьютер для того, чтобы мой партнер, у которого нет широкополосного подключения, мог перевести свой компьютер с XP на Linux с помощью моей машины.

Ричард Эйрз [Richard Ayres]

Я почти уверен, что дело не в *Firefox*. Похоже на то, что система все же цепляется за Ethernet-подключение. Наберите в терминале вот что:

```
route -n
```

Интересующая нас строка – последняя. Она начинаются с **0.0.0.0**, так как это шлюз по умолчанию для всех не-локальных подключений. Подозреваю, что она выглядит примерно так:

'0.0.0.0 192.168.0.1 0.0.0.0 UG 0 0 0 eth0'

Два последних числа во второй строке могут быть другими, но если она заканчивается на **eth0** (или что-нибудь другое, но не **ppp0**), то здесь-то и корень Вашей проблемы. вам необходимо убедиться в том, что любые настройки **eth0** вычищены из Вашей системы, особенно если вы больше не собираетесь пользоваться широкополосным подключением к своему провайдеру. Выделите это подключение в окне Network Settings (Сетевые настройки) и нажмите кнопку Delete (Удалить) (одна из трех малозаметных кнопок в верхнем правом углу окна).

Возможно также, что настройка вашего модемного подключения не завершена. Тот факт, что модем звонит, еще не означает, что подключение состоялось. Попробуйте запустить

```
sudo plog
```

после очевидно успешного модемного подключения. Просмотрите несколько последних строк протокола подключения: если что-то пойдет не так, вы сразу это поймете. Состояние сетевого интерфейса можно проверить и с помощью

```
/sbin/ifconfig -a
```

Если в выводе появился интерфейс **eth0**, то он не должен быть помечен **UP**, и у него не должно быть записей **inet addr**. А вот **ppp0** должен быть «поднят» (**UP**) и снабжен реальным адресом.

А может быть и такое: вы подключаетесь к провайдеру, но не можете разрешать доменные имена. Запустите в терминале следующие команды:

```
ping -c 5 www.google.com
ping -c 5 216.239.59.104
```

Первая из них – это попытка подключиться к Google по имени, а вторая минует DNS и запрашивает его прямо по IP-адресу. Если работает только вторая команда – значит, неверна Ваша информация о DNS. Необходимо связаться с провайдером и узнать у него адреса DNS-серверов, затем поместить их в **/etc/resolv.conf**. Это будет выглядеть примерно так:

```
nameserver 1.2.3.4
nameserver 1.2.4.5
```

Файл можно редактировать непосредственно либо воспользоваться инструментом Network Settings (Сетевые настройки). Возможно даже, что у вас там до сих пор прописаны имена доменных серверов широкополосного подключения. Их необходимо удалить. Если, проделав все это, вы все-таки не избавитесь от своей проблемы, попробуйте послать более подробные сведения, включая вывод вышеприведенных команд, в раздел Help наших форумов на **www.linuxformat.co.uk**. **НВ**

6 Установка Joomla

В Я ваш давний читатель, у меня двухгодичная подшивка *Linux Format* и Ubuntu 6.10. Но вот проблема, которую мне никак не решить: не могу устанавливать программы с вашего DVD.

Я несколько раз перечел статью об инсталляции из исходных текстов, читал следующие книги: *The Official Ubuntu Book*, *Beginning Ubuntu Linux* и *Ubuntu Unleashed*, но так и не понял своей ошибки. Может быть, вы сможете мне помочь, точно указав, что я должен вводить в терминале для инсталляции, скажем, *Joomla 1.0.11* с одного из ваших DVD?

Пер Неергорд Малер [Per Neergaard Mahler]

Вы выбрали весьма необычный пример, ведь *Joomla* – это web-приложение, которое необходимо устанавливать в каталог, доступный Вашему web-серверу. Кроме того, оно написано на PHP, скриптовом языке, следовательно, не нуждается в компиляции. Однако вот вам последовательность установки.

Сначала необходимо инсталлировать web-сервер. Как правило, это *Apache*, поэтому запустите диспетчер пакетов *Synaptic* и убедитесь в том, что **apache** и **libapache2-mod-php2** установлены. Протестируйте *Apache* вводом 'http://localhost' в адресную строку браузера.

Если все прошло гладко, приступайте к инсталляции собственно *Joomla*. Откройте терминал и наберите

```
sudo mkdir /var/www/joomla
sudo tar -xf joomla-1.0.11.tar.bz2 -C /var/www/joomla
```



› ***Joomla* обладает удобным web-инсталлятором, который сообщает, готова ли ваша система к установке программы.**

Вопрос-победитель (английская версия)

Безопасная связь

В Наш почтовый сервер обслуживается *Postfix*, а клиенты пользуются либо *Outlook Express*, либо серверным приложением *SquirrelMail*. Все работает отлично, и только клиенты нашей локальной сети могут обмениваться почтой с внешним миром.

Недавно мы приняли сотрудника, которому необходим доступ в Сеть со смартфона. Без проблем, мы открыли IMAP для его папок, и он может пользоваться SMTP-сервером *Postfix* для отправки сообщений – но только локальным адресатам (мы не хотим стать ретранслятором для спама). В идеале нам хотелось бы, чтобы этот сотрудник мог посылать сообщения не только по нашей сети. В какую часть *Postfix* нужно зайти, чтобы разрешить только ему обмениваться почтой с другими доменами изнутри **$my_networks**? Действующие почтовые правила должны оставаться неизменными.

Крейг Родуэй [Craig Rodway]

О Решение заключается в SMTP-авторизации, при которой пользователи проходят проверку перед отправкой сообщений. *Postfix* можно настроить так, чтобы он принимал сообщения только авторизованных пользователей.

Для авторизации *Postfix* использует *Cyrus-SASL*, поэтому убедитесь в том, что она установлена, а служба *saslauthd* запускается при загрузке системы.

Чтобы настроить *Postfix* на использование *Cyrus-SASL*, отредактируйте **/etc/Postfix/main.cf** и убедитесь в том, что mydomain, myhostname и mynetworks настроены правильно. Теперь добавьте в конец файла следующие строки:

```
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
broken_sasl_auth_clients = yes
smtpd_recipient_restrictions = permit_sasl_authenticated,permit_mynetworks,check_relay_domains
```

Четвертая строка необязательна. Она нужна для некоторых версий *Outlook Express* и *Microsoft Exchange*. Если ваш сотрудник будет пользоваться лишь смартфоном, то без нее можно обойтись. Перезапустите *Postfix* или заставьте его обновить конфигурацию – и любой зарегистрированный пользователь вашей сети сможет пользоваться SMTP-сервером отовсюду, настроив свою почтовую программу на использование SMTP-авторизации. Пользователи вашей сети смогут посылать почту, не меняя почтовых настроек.

По этому вопросу есть подробное HOWTO, доступное на http://Postfix.state-of-mind.de/patrick.koetter/smtpauth. Оно касается и вопроса использования TLS для шифрования подключений между пользователем и сервером. Это необходимо, так как в противном случае ваши пользователи будут вынуждены отправлять пароли открытым текстом.

SASL можно использовать и для авторизации изнутри сети. Например, можно настроить *Postfix* для школьной сети так, что все пользователи смогут обмениваться сообщениями внутри сети, но только учителя смогут отсылать почту вовне. **MC**

Первая команда создает каталог для установки *Joomla*, а вторая распаковывает в него *Joomla*-архив. Подразумевается, что **Joomla-1.0.11.tar.bz2** находится в текущей директории, иначе нужно вводить полный путь к архиву. Теперь наберите в браузере 'http://localhost/*Joomla*/' – начнется процесс инсталляции и настройки.

Как правило, первые шаги инсталляции для любого пакета, поставляемого в tar-архиве, одинаковы – распаковка архива и поиск файла, содержащего инструкции по установке, обычно README, INSTALL или что-нибудь столь же очевидное.

Журнальная статья об установке из исходных кодов применима к тем пакетам, что используют стандартный процесс установки из исходных текстов. Это относится к подавляющему большинству пакетов, но отнюдь не ко всем – что наглядно демонстрирует *Joomla*. **НБ**

7 Консольная почта

В Я хотел бы пользоваться почтой через оболочку командной строки. Не могли бы вы подсказать мне, как это сделать? Сейчас у меня Ubuntu 6.10.

jmullin, из форумов LXF

О вы хотите сказать, что вам нужен консольный почтовый клиент, встроенный в оболочку, или требуется отправлять почту с использованием скриптов оболочки? Существует несколько почтовых программ для терминала; наиболее популярная из них – *Mutt* (www.mutt.org). *Mutt* имеется в основном репозитории Ubuntu, и его можно установить через *Synaptic*.

Если вы хотите отправлять почту из *Bash*-скриптов, простейшее решение – команда *mailx* (и она, вероятно, в Вашей системе уже есть). Эта программа отправляет все, что получает со стандартного ввода, на указанный электронный адрес. Например:

```
echo "Hello World" | -s "Obvious example" me@example.com
```

Тема сообщения задается параметром **-s** (используйте кавычки, если в теме есть пробелы); все, что получено со стандартного ввода, составит тело сообщения. Для почтовой программы этого вполне достаточно. **MC**

8 Недо-гонки

В У меня Ubuntu 7.04 Feisty Fawn на ноутбуке Sony VAIO VGN-FJ250P. Этот дистрибутив устраивает меня почти полностью, исключая одну-две мелкие проблемы. Одна из них состоит в применении известного OpenGL. Кажется, он не поддерживается моей Linux-системой. Я уверен, что видеочипсет Intel ни при чем, так как у меня двойная загрузка с Windows XP Pro, и там приложения с OpenGL работают прекрасно.

Одно из таких приложений – это *Planet Penguin Racer*, игра, которая хорошо идет с Live CD, но отказывается работать, будучи установленной на жесткий диск. Попытка запустить ее из меню ни к чему не приводит, а запуск из командной строки вызывает появление следующего сообщения:

'*** ppracer error: Couldn't initialize
video: Couldn't find matching GLX
visual (Success)
Segmentation fault (core dumped)'.

Джим Смит [Jim Smith]

О Хорошо, что OpenGL работает на Вашем оборудовании с Live CD: это говорит о том, что видеокарта поддерживается, а необходимое ПО присутствует на Live CD.

Почти наверняка это проблема настройки, кроющаяся в **xorg.conf**. Вызвана она тем, что инсталлятор неверно настроил Вашу видеокарту.

Загрузитесь с Live CD, смонтируйте один из разделов жесткого диска или USB-накопителя и скопируйте туда **/etc/X11/xorg.conf**. Затем загрузитесь с жесткого диска и сравните «местную» копию **xorg.conf** с только что сохраненной.

Вероятнее всего, дисковая версия файла использует не тот драйвер (строка **Driver** в разделе **Device**) или не загружает GLX-модуль. Прежде чем менять что-либо в этом файле, сделайте резервную копию: вы ведь не хотите усугубить ситуацию.

«Правильный» драйвер для Вашей карты – i810, хотя сработает все, что будет найдено в версии файла с Live CD. Модуль GLX загружается включением следующей строки в секцию **Module** **xorg.conf**:

```
Load "glx"
```

Если обе эти проблемы устранены, а OpenGL все же не работает, следует пройтись по обеим версиям файла и выяснить, чем они отличаются. А можно просто заменить дисковый вариант версией с Live CD, ведь она рабочая. **НВ**



› **Для работы OpenGL-программ типа *ppracer* нужно правильно установить и настроить 3D-графические драйверы.**

9 Непостоянство dev

В В одном из ваших прошлых номеров рассказывалось, как настроить GPRS-соединение в Linux. Помимо прочего, там предлагалось создать символьную ссылку вроде **ln -s /dev/ttyACM0 /dev/modem**. Действительно, это помогает. Но после перезагрузки ссылка исчезает. А недавно я столкнулся еще и с тем, что Nero Linux 3 требует доступа к устройству **/dev/sg0**. Выставляю права, но после перезагрузки они опять сбрасываются: как я понимаю, это происходит из-за динамической природы **/dev**. Собственно сам вопрос: а как сделать так, чтобы после перезагрузки не приходилось вновь производить вышеуказанные манипуляции?

Евдокимов А.В.

О Динамическим каталогом **/dev** в Linux 2.6 заведует программа *udev*. В Сети доступно обширное руководство по написанию правил *udev* (правда, на английском языке) - http://reactivated.net/writing_udev_rules.html. Чтобы сделать символьную ссылку **/dev/modem** постоянной, можно использовать правило:

```
KERNEL=="ttyACM0", SYMLINK+="modem"
```

Для sg0 можно попробовать следующее:

```
KERNEL=="sg0", MODE="0666"
```

Правила *udev* добавляются в специальный файл – обычно он называется **/etc/udev/rules.d/10-local.rules**. Два последних компонента пути могут зависеть от дистрибутива, но "local" и "rules" в имени файла присутствуют обязательно. Не вносите собственные правила в другие файлы *udev* – это может привести к проблемам. **ВС**

10 Гауссу посвящается

В У меня возникла следующая задача, которую, как мне кажется, решить довольно легко. Необходимо дать всем файлам в одном каталоге случайные названия (можно и цифровые), но сохранить расширение. Хотелось бы, чтобы это был простенький скрипт на *Bash* или Perl. Дело в том, что мне подарили MP3-плеер, который умеет либо проигрывать мелодии подряд, либо работать в режиме случайного выбора песни. Оказалось, что датчик случайных чисел в этом плейере плохой и постоянно выдает фактически одну и ту же последовательность песен, и, как следствие, режимы мало отличаются друг от друга. Поэтому вместо встроенного датчика (и вместе с ним) я хочу использовать более качественный компьютерный.

Маслий Алексей

О Вашу задачу можно решить буквально одной командой:

```
$ for oldname in *.mp3; do mv $oldname $RANDOM.mp3; done
```

Переменная оболочки **RANDOM** возвращает случайное число в диапазоне от 0 до 32767, которое генерируется Bash при каждом ее прочтении. Поскольку значения **RANDOM**, в принципе, могут повторяться, следует учесть этот факт во избежание затирания одного файла другим:

```
for oldname in *.mp3; do
  while [ -z $newname -o -e $newname ]; do
    newname="$RANDOM.mp3"
  done
  mv $oldname $newname
done
```

Цикл **while** генерирует новое имя для каждого файла до тех пор, пока не найдет свободное (**-e $newname** проверяет существование файла с именем **$newname**). Условие **-z $newname** необходимо, чтобы цикл инициализировал значение переменной newname перед началом работы.

Изменив **$RANDOM.mp3** на **$1$RANDOM.mp3**, вы получите возможность указать общий префикс (например, **"song_"**) в качестве параметра командой строки скрипта

```
$ ./randomizer.sh song_; ls *.mp3
song_10306.mp3  song_123.mp3  song_20096.mp3
```

Скрипт несложно обобщить и на случай, когда файлы в каталоге имеют различные расширения: достаточно воспользоваться конструкцией **ext=${oldname##*.}** для его извлечения и заменить **1RANDOM.mp3** на **1RANDOM.$ext**. Да, и не забудьте изменить фильтр ***.mp3** в первой строке на что-то более подходящее, скажем ***.{mp3,wav}**. **ВС**

Нужна помощь!

› Для наилучшего ответа на ваш вопрос нам нужно знать как можно больше подробностей. Детально опишите конфигурацию системы. Если вы получили сообщение об ошибке, приведите текст сообщения и точно опишите вызвавшие его действия. Если у вас проблемы с оборудованием, то опишите его. Если Linux уже запущен, то выполните в root-терминале следующие команды и прикрепите к письму файл **system.txt**:

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

› Пожалуйста, помните, что сотрудники журнала НЕ являются авторами или разработчиками Linux, любых пакетов или дистрибутивов. Зачастую люди, отвечающие за приложения, выкладывают большую часть информации на web-сайты. Попробуйте почитать документацию!

Мы стараемся ответить на все вопросы. Если вы не нашли ответ на свой, это, возможно, потому, что мы уже ответили на похожий вопрос.



! Вопрос-победитель (русская версия)

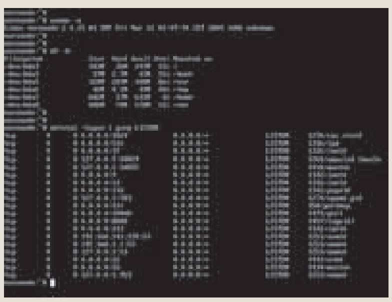
Автор следующего вопроса, скрывающийся под псевдонимом ***asg128***, получает подарочный сертификат на 1000 рублей от интернет-магазина LinuxCenter.Ru! Просим победителя выйти на связь с редакцией: info@linuxformat.ru

★ Пожиратель трафика

В Как узнать, какие программы используют подключение к Интернету (или к любому другому сетевому интерфейсу)? Я использую Mandriva 2007, в Интернет выхожу через модем. Если верить окну «Статистика КРРР», что-то закачивается на мой компьютер, даже если я не запустил ни одного приложения.

asg128

О Быть может, ваши опасения и необоснованны – протокол PPP предусматривает обмен служебными пакетами между вашим компьютером и провайдером, поэтому незначительный трафик может существовать даже при отсутствии «полезной» нагрузки: закрытом браузере, почтовом клиенте и т.п. Не исключено также, что Mandriva просто загружает списки обновлений. В любом случае, проверить, какие программы используют соединение с Интернетом, можно командой *netstat -ap*, желательно – из root-терминала. Найдите в ее выводе секцию *"Active Internet Connections"*: PID и имя программы отображаются в последней колонке. Имейте в виду, что если вам действительно не повезло подцепить rootkit, он, скорее всего, не появится в выводе *netstat*. **ВС**



Большой вопрос

На этом раздел заканчивается, но наши эксперты не расходятся по домам до тех пор, пока не дойдут до сути общей проблемы.

11 Раздвоение рабочего места

В Только что приобрел новый LCD-монитор на замену старому CRT. У моей видеокарты Nvidia есть два видеовыхода – так, значит, можно подключить к карте оба монитора и растянуть мой рабочий стол KDE на них? У меня Gentoo 2006.1, видеокарта Nvidia FX5200.

Гарри Лич [Harry Leach]

О Ответ: да. Существует стандартный метод комбинации двух экранов в один X-дисплей, называемый *Xinerama*. Обычно он используется с двумя видеокартами, но у Nvidia есть драйверы, обладающие функцией под названием *TwinView*, которая позволяет отображать два экрана с одной видеокарты, каждый на отдельном выходе, сохраняя при этом совместимость с *Xinerama*.

Настроить *TwinView* (и *Xinerama*) несложно, при условии, что у вас есть фирменные драйверы Nvidia, а не свободный драйвер *nv*. Сначала убедитесь в том, что *X* работает с *Nvidia-драйвером*: самый очевидный показатель этого – появление логотипа Nvidia при запуске *X*. Затем запустите *nvidia-settings* из терминала root. В Gentoo это отдельный пакет, поэтому найдите его, если еще этого не сделали. Выберите X Server Display Configuration (Настройка дисплея X-сервера) в списке слева – вы должны увидеть оба дисплея, хотя один из них может быть помечен как отключенный. Если один из дисплеев недоступен, нажмите кнопку Detect Displays (Найти дисплеи), выделите дисплеи по очереди и установите для каждого требуемое разрешение. Лучше выставить одинаковые разрешения экрана, но любой оконный менеджер, понимающий *Xinerama*, управится и с разными разрешениями.

Теперь настройте Position (Расположение) для каждого экрана. Для максимальной управляемости можно делать это в абсолютных величинах, но лучше бы настроить дисплеи так, чтобы один находился Right Of (Справа) от другого, а второй, соответственно, слева. Нажмите кнопку Save To X Configuration File (Сохранить изменения в файле настройки X), выйдите из учетной записи и перезапустите *X*.

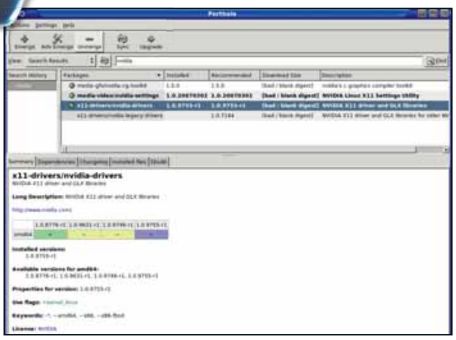
Вы должны увидеть рабочий стол, растянутый на два монитора, но для окончательной доводки понадобится еще кое-что. Необходимо убедиться, что все приложения скомпилированы с поддержкой *Xinerama*. Если **xinerama** еще нет во флагах USE, то отредактируйте **/etc/make.conf** и добавьте его туда, затем скомпилируйте заново нужные вам программы:

```
emerge --update --deep --newuse --ask world
```

Конечно, это долго, но зато по окончании процесса можно перезапустить KDE и приступить к настройке дисплеев по своему вкусу. Так, рабочий стол можно украсить едиными обоями, растянутыми на два экрана – или использовать разные для каждого монитора. Панель задач оставить на одном экране – или растянуть на оба. Раздел Multiple Monitors (Мониторы) в настройках рабочего стола позволяет указать поведение окон и выбрать дисплей по умолчанию для открытия новых окон. Найдите в этом разделе свой LCD-монитор.

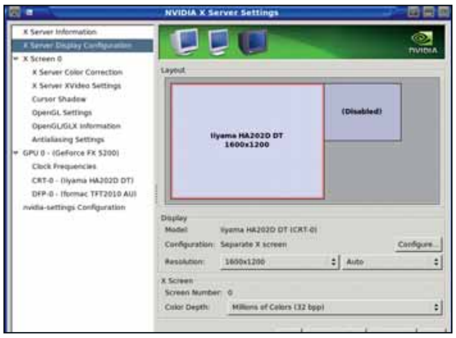
Может пригодиться параметр Special Window Settings (Особые оконные настройки) меню Advanced (Дополнительно), доступного по правому щелчку на заголовке окна. Он позволяет изменить стандартный порядок работы с окнами приложений. Удобно и на одном дисплее, но особую актуальность приобретает на двух. Например, я открываю «холст» *GIMP* на одном экране, а панель инструментов – на другом, чтобы не мешала редактировать изображение. НБ LXF

Шаг за шагом: Объединение двух экранов в один дисплей



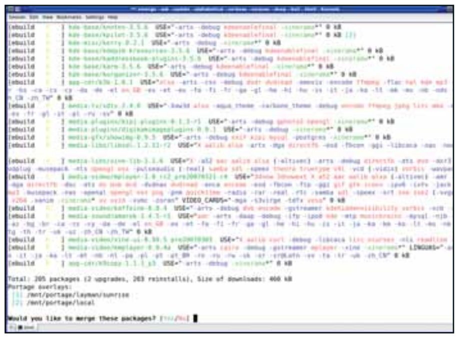
1 Инсталляция

Чтобы воспользоваться *TwinView*, нужно сначала установить драйверы Nvidia. Пакет *nvidia-settings* тоже пригодится.



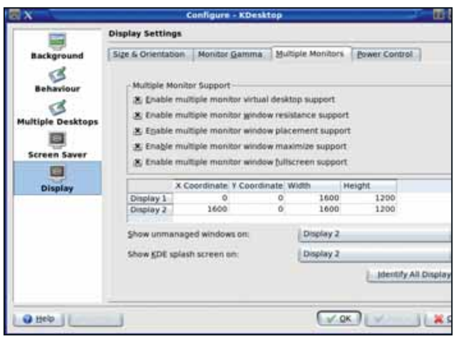
2 Установка разрешения экранов

Настройте *TwinView* для своих мониторов с помощью *nvidia-settings*. Экранные разрешения могут быть разными, и все же: чем меньше разброс, тем лучше.



3 Перекомпиляция

Если ваша система скомпилирована без использования USE-флага **xinerama**, то вам предстоит длительный сеанс перекомпиляции. Тем временем можно продолжать работу в системе.



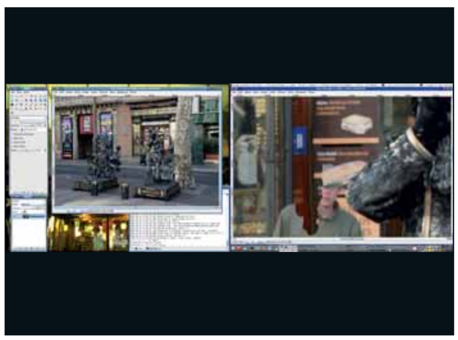
4 Настройка

У KDE есть средства управления несколькими мониторами. Например, максимизированное окно может быть открыто только на одном мониторе.



5 Поведение

Можете использовать свои мониторы как один большой дисплей или как два отдельных экрана. Здесь, у каждого из мониторов свои обои, а панель задач отображается лишь на одном из них.



6 Добавочное пространство

Рабочий стол с разрешением 3200x1600 – это уйма рабочего пространства. Если вы работаете с несколькими окнами одновременно, производительность вашего труда может резко возрасти.



Лучшие новинки открытого ПО на планете

LXF HotPicks



Ник Вейч
Чудо-математик, фанат Flickr, иногда редактор, Ник из высокой башни LXF краем глаза осматривает поле свободных программ в поисках хорошеньких приложений.

В ЭТОТ РАЗ ТОЛЬКО ДЛЯ ВАС: Semantik » Sylpheed » UADE » Gramps » Warzone 2010 » Kwa » Fracplanet » Speedcrunch » MyMP3db » Viking

ПО для структурирования идей

Semantik

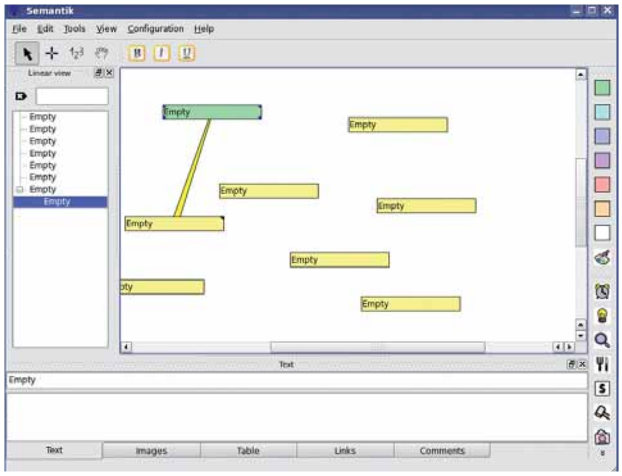
Версия 0.28 **Сайт** http://freehackers.org/~tnagy/kdissert.html

Есть мысль, значит, нужно создать карту разума (mind map). Эти паукообразные диаграммы великолепны для мозгового штурма, но не всегда приводят к законченной работе. *Semantik* похож на инструменты майнд-мэппинга в том, что он предназначен для упрощения отслеживания мыслительного процесса и связи идей путем их структурного документирования, но его авторы заявляют, что он отличается от других, так как фокусируется на создании документации, а не только красивых диаграмм.

Мысли можно упорядочить в виде дерева, редактировать, связать вместе, разобщить и вообще делать с ними что угодно, пока они не встанут на место некой структуры, хорошо описывающей ...... (тип документа вставьте сами: диссертацию, руководство, книгу, ...).

Согласно целям программы, она позволяет создавать только жесткие древовидные структуры. Вы можете вывалить кучу различных идей и связать их вместе, но не можете создать петлю или потомка с несколькими предками – в любом случае, это может сработать для напечатанной книги?

Приложение не ограничено только текстом. Будьте троюродным кузеном дьявола и набросайте в него несколько картинок, а то и чисел ('8' обычно любимое число в таких делах)? Каждый узел может иметь связи, что весьма удобно для ссылок, сносок и тому подобного; также, по желанию, вы можете добавить причудливые таблицы. Имеется и набор небольших иконок для присоединения их к объектам. Если вы хотите напомнить себе, что некая мысль осенила вас после небольшого кусочка сыра, то почему бы не сделать это при помощи картинки?



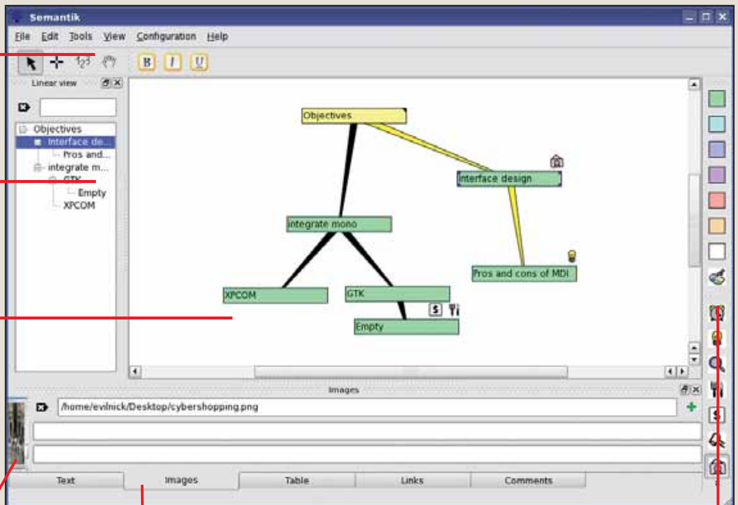
› **Предварительная карта разума одного из сотрудников *Linux Format*.**

Наконец, когда работа сделана, вы можете вывести ваш опус в LaTeX-подобном формате для экспорта в PDF или просто в HTML. Вероятно, программе не хватает лишь одного: на данный момент невелик выбор форм представления конечного результата, хотя при желании вполне можно провести постобработку и эффективно переформатировать дизайн, вставив немного CSS-кода, если вы имеете к этому склонность.

Semantik не вполне готов как комплекс обработки мысль-в-документ; просто инструменты работы с текстом (на данный момент) весьма ограничены для этого. Но для документирования потоков мыслей, отслеживания связей и составления скелета для дальнейших действий по документированию он, похоже, работает.

Если он напомнит вам инструмент под названием *KDissert*, который давно тут отирался, то вы очень проницательны. На самом деле, это и есть *KDissert*, но немного измененный и с новым именем. Графический интерфейс переработан, но данная версия имеет сходную с оригиналом функциональность, так что вы не сочтете его слишком трудным в настройке. Теперь вперед, к созданию карты разума – и постарайтесь не заблудиться!

Исследуем интерфейс Semantik



Панель инструментов
Здесь не так уж много инструментов – но много вам и не надо!

Создать дерево
На этой панели отображается древовидная структура вашего документа.

Переместить и связать узлы
Вы можете переносить и размещать узлы по всему создаваемому документу и добавлять связи предок/потомок.

Добавить изображения
Полезные снимки и диаграммы могут быть привязаны к узлам, как напоминания или как более детальное описание.

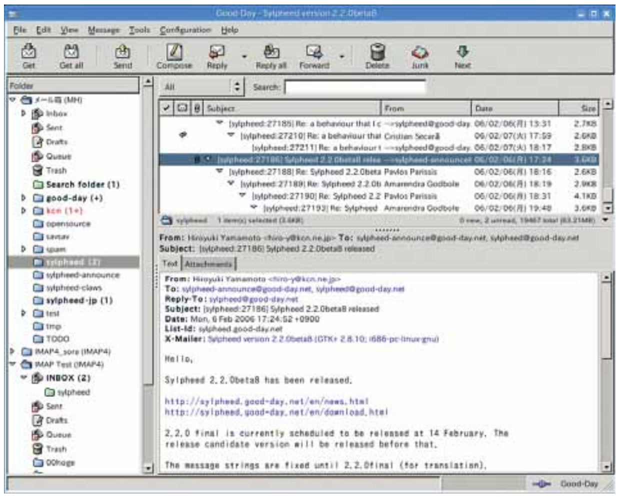
Добавить данные
Для добавления таблиц, связей и ссылок, равно как и простого текста, щелкайте на вкладках.

Цвета и иконки
Если нужно, выберите различные цвета для ваших узлов и добавьте полезные иконки-напоминания.

Почтовый клиент

Sylpheed

Версия 2.4.0 **Сайт** http://sylpheed.sraoss.jp/en



В этом пункте, наверное, должна быть удивительная история или волнующий рассказ на какую-то якобы отвлеченную тему, затем чертовски ловко и умно переходящую к аналогии с данной программой. Увы, почтовый клиент – это нудно, так что не в этот раз. Облом.

Так-то так, но ведь и все, что вы запускаете каждые десять минут весь рабочий день напролет, вероятно, кажется нудным – зато если оно не выполняет того, что вам нужно, или работает плохо, тут уж не до скуки: это сущая заноза в заднице, что медленно доводит вас до безумия и превращает в издерганного психопата. Вскоре вы... а, вы вообще-то хотели узнать о *Sylpheed*. Ну, раз так надо…

Sylpheed может показаться немного странным в мире, где царят *Outlook*, *Thunderbird*, *Mail* от Apple и прочие виды почтовых клиентов с хитрыми расширениями и тому подобным. С виду *Sylpheed* – натуральный простачок. Но банальные панели GTK-интерфейса скрывают истинную функциональность невязчивого, но хорошо сконструированного почтового клиента.

Простое трехпанельное окружение содержит все, что нужно знать. Базовые функции все здесь: GnuPG-защита, проверка орфографии, исчерпывающая система фильтрации (включающая сохраненный поиск), опция управления несколькими ящиками и что там еще вам может понадобиться.

Вдобавок *Sylpheed* весьма расширяем. Другие почтовые программы могут иметь дополнительные модули для выполнения специальных задач, но что сравнится с возможностью легко добавить свой собственный? При необходимости дальнейшей обработки *Sylpheed* способен передавать различные параметры почтовых сообщений внешним командам (одна из возможностей – углубленная проверка на спам).

> «Передает внешним командам параметры почтовых сообщений для обработки.»

› На вид он простоват, но что вы хотите от почтового клиента?

Этот новый стабильный релиз программы завершил множество улучшений, сделанных в кодовой базе за последние три-четыре месяца, включая подстройку интерфейса, множество исправлений и расширенную фильтрацию спама (очень полезно в современном мире).

Мы уверены, что вы не обремените себя попыткой его использования. Почтовые клиенты – это такая штука, которую люди используют просто по привычке. Но вам следует дать *Sylpheed* шанс. Он сумеет вам понравиться.

Эмулятор DeliTracker

UADE

Версия 2.0.7 **Сайт** http://zakalwe.fi/uade



Когда мы говорим, что «миллодии и ридмы», звучавшие в демо-заставке Amiga, были неподражаемы, мы не имеем в виду, что людям более не создать подобной музыки. Мы хотим сказать, что она была скрипучей, грязной и необработанной. Но некоторые из самых великих и захватывающих мелодий, которые нам доводилось слышать – они, вероятно, все еще эхом гуляют во внутренних пещерах разума Майка – были созданы именно в тот период, и с *UADE* мы вам это докажем.

Если вы, как и большинство команды *Linux Format*, вообразили, что мы просим достать старый A1200 с чердака только чтобы услышать несколько треков Йохена Фельдкеттера [Jochen Feldkötter], то расслабьтесь: этого не надо. Благодаря эмуляторам, ностальгия – уже не то, что осталось в прошлом. Кому нужны SoundFont'ы, MIDI или цифровая обработка сигналов, если вы можете насладиться компьютерной музыкой, просто слушая ее – во всем ее 8-битном великолепии?

Для воспроизведения старых треков Amiga вам необходимо установить превосходный UAE и слушать их из родной среды, но если даже это слишком утомительно, то, возможно, вы захотите потратить минут десять на установку *Unix Amiga Delitracker Emulator*.

Приложение *UADE* во многом основано на реальном ПО Amiga – *DeliTracker* и *Eagleplayer*, так что не опасайтесь неверной интерпретации исходной музыки (обе программы – те самые, из Amiga). Каждый тонкий нюанс именно таков, как должен быть. Вообще-то 8-битный звук не содержит тонких нюансов: точнее будет сказать, каждый скрежещущий грубый нюанс именно таков, как и должен быть.

Вы можете запускать *UADE* прямо из командной строки с обычным скопищем ключей, но намного проще использовать модуль XMMS, он полностью интегрирует демо-треки Amiga в вашу коллекцию Arctic Snow Monkey Patrol.

> «Верите или нет, но люди и вправду все еще делают треки в этом формате.»

› Это звук с разрядностью менее 16 бит.

Если в этой коллекции нет оригинальных файлов, кое-какие примеры можно взять из дистрибутива программы; кроме того, на домашней странице *UADE* имеется множество ссылок, которые стоит посмотреть. Верите вы или нет, но люди все еще создают треки в этом формате, так что все это не почило в прошлом.

И еще одно напоследок. Если вы никогда не видели демо-сцен Amiga и подумали, что программа *UADE* не работает, потому что извергает жуткий режущий ухо скрежет, не суетитесь: это и есть штатный режим.

Генеалогическая программа

Gramps

LINUX FORMAT HotPicks *повторный визит*

Версия 2.2.7 **Сайт** www.gramps-project.org

По каким-то причинам люди очарованы своими предками. В великой схеме знания, кто мы и куда идем, возможно, это помогает определить точку на огромной карте жизни Google, откуда мы явились. Очевидно, все мы происходим от одних и тех же первобытных людей, которые, как полагают ученые (именно так), скитались по саванне и вели трудную жизнь кочевых охотников в поисках пищи. Когда примерно это имело место, зависит от конкретной личности; что касается Майка, то мы полагаем, где-то в прошлый четверг.

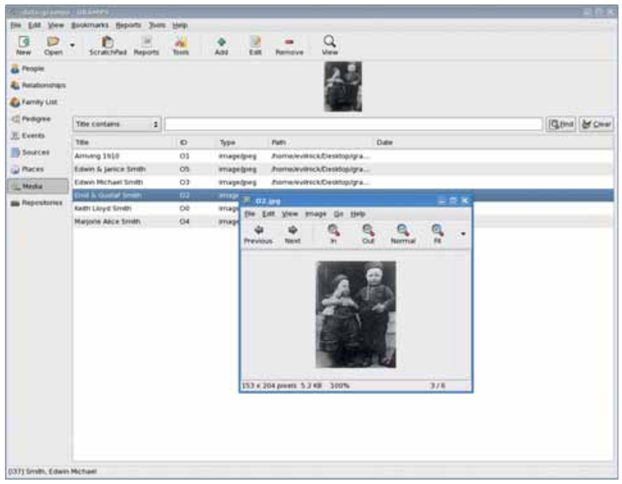
Но если вы хотите узнать, были ли в вашей родне конокрады, бандиты с порубежья, военные преступники, знаменитые ученые или, что более вероятно, безобидные люди, не совершившие ничего выдающегося, придется как-то разобраться с тем, что вы нарыли. По существу, вам нужна некая настраиваемая база данных, помогающая управлять персональной информацией: рождения, смерти, браки – и, возможно, хранить несколько картинок и диаграмм.

Хорошая новость: вам не нужно охотиться за таким инструментом в саванне, потому что *Gramps* уже существует. *Gramps* вообще-то уже рассматривался в HotPicks ранее, в LXF70, а до этого в LXF56, но с того времени он значительно улучшился: появился новый интерфейс и множество дополнительных опций добавления информации. Кроме того, он настолько хорош, что о нем стоит поговорить вновь. Это действительно так, поверьте.

Хотя вы можете использовать *Gramps* лишь для указания родства и рисования генеалогического древа, в нем заложено куда больше. Когда вы покончите с вашей ближайшей родней, у вас, вероятно, возникнут затруднения насчет, к примеру, дат рождения (вообще-то трудности могут начаться задолго до этого этапа). Вам может понадобиться позвонить тетушке Этель, чтобы уточнить, действительно ли прадедушка Джордж вернулся с войны и объявил, что женат на жирафе. С помощью *Gramps* вы можете не только хранить связи между информацией, но и отмечать, откуда она появилась: в конце концов, тетушка Этель уже и день недели с трудом припоминает, а не то что события, случившиеся десятилетия назад. В будущем вполне может оказаться, что некие данные выпадают из общей картины, и потому всегда полезно знать степень достоверности информации.

Подобные особенности показывают истинную силу *Gramps* – это нечто большее, чем инструмент для рисования генеалогического древа: это приложение для ведения истории семьи, которое может помочь исследовать темное прошлое вашей родни сквозь века.

Кроме заметок, *Gramps* также хранит информацию другого сорта. Коли вы собрали полную историю жизни каждого персонажа вашего древа, почему бы не добавить пару портретов или сканированных документов по соответствующим событиям?

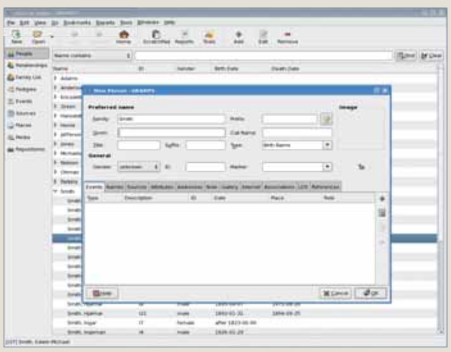


› ***Gramps* не просто рисует генеалогического древо: он может создать историю вашей семьи.**

Главный козырь этого приложения – способ связывания информации. Если вы соединяете двух людей вместе, поскольку они поженились, они автоматически становятся доступны в наборе родителей для любых детей, которых вы, возможно, захотите добавить. Предусмотрены все возможные ситуации, включая необычные: Майку наверняка понравится, что *Gramps* способен разрулить проблему при отсутствии точной информации про пол (или, возможно, вид) предка.

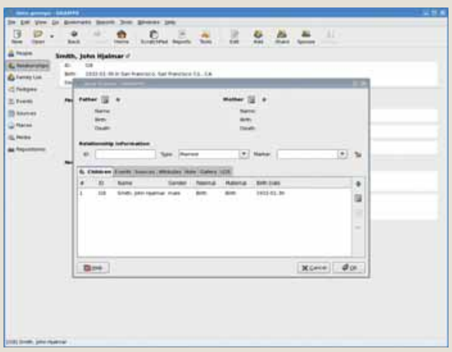
> «Поможет исследовать темное прошлое вашей родни сквозь века.»

Шаг за шагом: Создание генеалогического древа



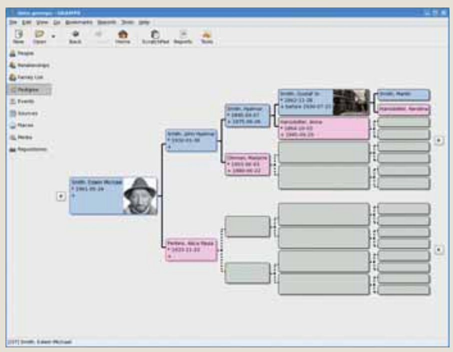
» Добавьте людей

Сначала создайте базу данных – щелкните на кнопке New [Создать] и выберите расположение. Затем нажмите Add [Добавить] для добавления нового лица и ввода деталей.



» Обозначьте связи

Чтобы пополнить данные о вашей семье, выберите персону и щелкните на строке Relationships [Родство] слева. Нажмите Add [Добавить], чтобы добавить его родителей.



» Генерируйте!

Добавьте всех, кого вспомните, затем выберите Pedigree [Родословная] в левой панели – и увидите генеалогическое древо!

HotGames Развлекательные приложения

Лазерно-стратегически-смертельная возня

Warzone 2100

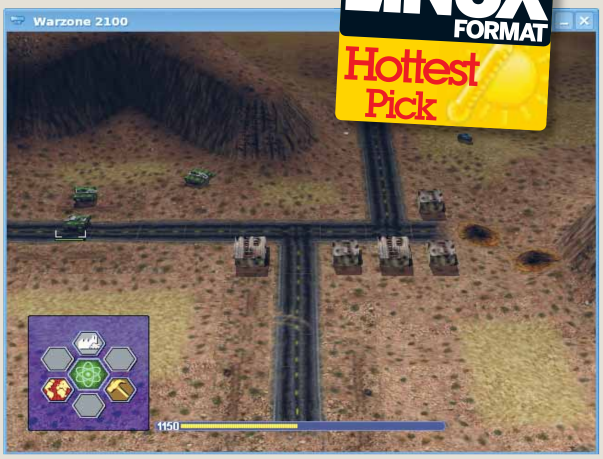
Версия 2.0.6 **Сайт** http://wz2100.net

Недалекое будущее; известный мужской джаз-бэнд вернулся из отставки, на этот раз не для песен, а для битвы. На танках. Вот практически и все. Правда, мужской джаз-банд тут ни при чем. Зато в *Warzone 2100* есть танки, и много.

Много лет назад игра под названием *Dune II* объявила миру о возникновении нового игрового жанра, и мы все хотели играть в нее, черт возьми. Пост-фактум, этот жанр получил название RTS – стратегия реального времени, и с того времени игроки неистово пытаются управлять мириадами ресурсов (специи, руда, дерево, сласти или что-то еще), чтобы строить вещи (вообще-то преимущественно танки) и не давать другим людям (тоже с танками) забрать их сладости (или что-то еще).

Warzone не претендует на великую идею, отличающую ее от предыдущих игр – точнее, она вполне укладывается в разряд *Command & Conquer* – но имеет несколько приятных дополнительных элементов. Сюжет завязан вокруг исследований и освоения территорий. Каждая миссия имеет свою цель: заполучить некий артефакт, построить базу и так далее. Путь вам преграждают злобные типы, имеющие собственные технологии и хитрые планы. Налицо все ожидаемые элементы: технологии, чтобы исследовать, и ресурсы, чтобы их найти и использовать. Технологии позволяют разрабатывать новые типы машин (да, вы имеете возможность проектировать их самостоятельно для лучшего использования ресурсов). Первые уровни предельно просты, с явным преобладанием числа, достаточного для завоевания победы, но позднее вы должны будете делать сложный выбор как использовать ресурсы и как снаряжать свои войска.

> «Потом вам предстоит трудный выбор: как развернуть свои войска.»



› **Это работа для танка! Что ж – здесь их по-настоящему много!**

Способ управления в *Warzone 2100*, возможно, поначалу будет трудноват, и иногда хочется более ясного отображения карты, но в целом графика, звук и игровой процесс на высоком уровне – вам определенно необходимо попробовать ее. Крушите!

Симулятор пожаротушения

Kwa

Версия 0.2 **Сайт** www.ulrichbochert.de

Побывавшие на второй выставке Libre Graphics в Монреале, возможно, выкажут больше, чем мимолетный интерес к этому горячему-горячему HotPick. *KWA* – это стратегия, вручающая вам святую и тяжкую ответственность за защиту от пожаров лесистой и прекрасной девственной природы Канады.

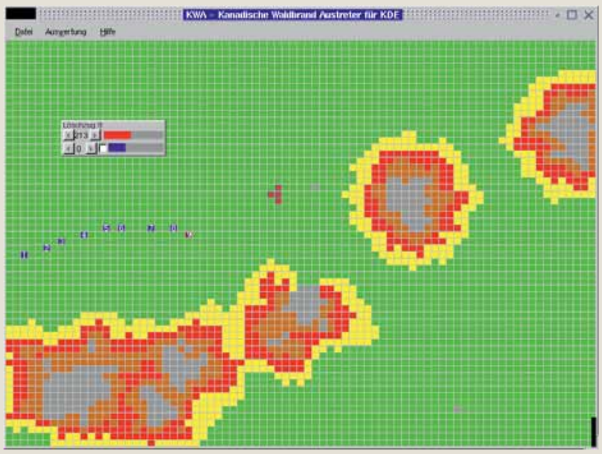
У вас под началом – команда пожарных. Как у реальных экземпляров, у них хватает ума тушить пламя у себя под ногами, но они, в отличие от вас, не способны озирать окрестные кленовые леса.

Щелчок по числу на экране приводит к выбору отдельного пожарного подразделения. Повторный щелчок устанавливает ему цель для тушения. Вам следует торопиться, ибо огонь может быстро выйти из под контроля, но не делать при этом глупостей.

Главное – защитить деревни (расположенные на карте случайным образом), но будьте также готовы к возникновению огня практически в любом месте карты.

Ко всему хорошему, до возвращения на базу вода и топливо у пожарных лимитированы. Правый щелчок позволяет вывести дополнительную информацию. Направление и скорость тут не самое важное: красная и синяя полоски показывают остаток топлива и воды, и их нужно контролировать, чтобы не бросить свои подразделения в чащу леса с пустыми руками, желая погасить последний большой пожар. Вы в любой момент можете увидеть результат ваших действий, показывающий, какие ошибки вы наделали, и сколько вековых деревьев рухнуло, подвергшись внеплановому окислению.

> «Ко всему хорошему, вода и топливо у пожарных лимитированы.»



› **Оранжевая капля? Это пылающий ад...**

Это не та игра, в которой можно что-то выиграть, но интересно поэкспериментировать с различными стратегиями и угадать, где чиркнет очередная неосторожная спичка. Если вам повезет, то пойдет дождь. Но при любом раскладе вынесение соответствующего вашим подвигам карающего вердикта неотвратимо.

Генератор геофракталов

Fracplanet

Версия 0.3.3 **Web** www.bottlenose.demon.co.uk/share/fracplanet

За свои утомительные заморские поездки с целью посетить хакеров всех континентов, команда LXF наследила углем вполне достаточно, чтобы снабдить Китай карандашами на последующие 10 лет. Однако не все потеряно. Пока знакомый нам мир погибает в катаклизме пластиковых отходов и отработанного машинного масла, мы займемся случайными мирами, сотворенными во *Fracplanet*!

Сразу скажем: это не соперник для *Bryce* в Windows/Mac, или даже (если вы злопамятны) *VistaPro*. *Fracplanet* легко создает фрактальный глобус, заполненный сушей, горами, океанами, береговыми линиями и полярными ледниками, для желающих вспомнить добрые старые деньки. Главное окно отображения имеет несколько простых управляющих элементов, позволяющих настроить вращение Мира в полноцветном великолепии и наблюдать смену дней. Если вы очарованы вашим творением, то можете даже попытаться облететь его, хотя точно управлять мышью не так-то просто. Требуются умелые руки, чтобы избежать превращения дивной прогулки среди заснеженных пиков горных массивов в римейк *Alive*.

Ясно, что в осмотре готовой планеты мало радости. Хочется узнать, что произойдет при создании полностью ледяного мира или планеты множества островов, какие вам случалось видеть в *Civilization*, если вы играли достаточно много. Не отчаивайтесь, обо этом тоже позаботились; вам просто нужно покрутить регуляторы. Панель управления имеет подсказки, и если вы немного придержите мышь над кнопками, то появится полезный совет; но все и так предельно просто – вам лишь нужно поэкспериментировать, чтобы увидеть, что происходит при изменении настроек.

> «Уж лучше заранее представить себе верхние слои атмосферы.»



› И это созданный нами Мир?

Да тут еще и облака! Честно говоря, с ними придется попыхтеть – уж лучше заранее представить себе верхние слои атмосферы.

Если вы полюбили свой маленький новый Мир, можно уговорить *Fracplanet* экспортировать вашу модель для использования в POVRay, или *Blender*, или любой другой программе, поддерживающей стандартные форматы объектов. Вы также можете сохранить плоские текстуры в формате PPM, который легко преобразуется во множество других графических форматов, если впоследствии они понадобятся вам для использования в вашем любимом 3D-редакторе.

GPS-менеджер

Viking

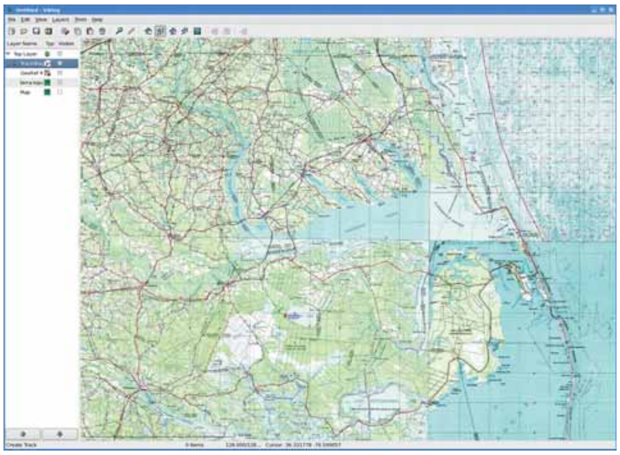
Версия 0.1.2 **Сайт** http://sourceforge.net/projects/viking

Заблудиться – штука неприятная. Конечно, есть ‘пропавшие в музыке’, что, видимо, нормально, а может, даже мило. Что еще? Заблудиться во Франции? Нет, спасибо. Еще заблудшие? Затерялись в *Lost*? В *Lost* все теряются, вот в чем дело. Потерялись не они, потерялись зрители. Так или иначе, в целом, я думаю, вы согласитесь, что за некоторыми исключениями, которые мы попытались обрисовать, заблудиться – неприятно. Вот почему природа дала нам спутники на геостационарной орбите и GPS-устройства.

В наши дни полно доступных гаджетов, сообщающих вам, насколько медленно вы взбираетесь на высокую гору, или даже (что более полезно) где ближайший бар. Эти устройства могут также записывать большой объем информации о том, где вы были и что вы делали.

Управление подобными данными обычно основано на куче проприетарного хлама, поставляющегося с вашим GPS-устройством, и это позор, потому что практически все устройства используют одни и те же стандарты для записи информации. Проект *Viking* – попытка сделать ваши геоданные свободными – путем построения многослойных комплексных карт на любой случай. Приложение позволяет импортировать данные со многих источников, и даже сумеет загрузить хорошие карты текущей местности с различных онлайн-ресурсов (хотя, если честно, иногда это невозможно). Знайте, что поддерживаемые источники не охватывают весь мир, так что в местах, куда вас завез ваш горный велосипед в выходной, бесплатной карты может не оказаться.

Эта версия *Viking* намного стабильнее предыдущих. Хотя это вылилось в отсутствие многих функций импорта и экспорта данных, он все же умело работает с интересной информацией из других источников! В отличие от других GPS-менеджеров для Linux, которые в данный момент похоже сникли, этот проект демонстрирует устойчивое развитие.



› Крестиком обозначен Грэмов недоеденный сэндвич с ветчиной.

> «Можете автоматически нашпиговать вашу карту тайными кладами.»

Если вы кладоискатель (и у вас установлен *Gcget*), можете автоматически нашпиговать вашу карту захоронками, полными всякой ерунды. Если вы хотите потратить несколько часов на тщательное разбирательство, какие тайники команда *Linux Format* забивает невостребованными футболками и прошлыми номерами – вероятно, вам следует дать шанс *Viking*’у.

Потоковое вещание музыкальной коллекции

MyMP3db

Версия 3.0 **Сайт** www.dontpokebadgers.com/MP3db

MyMP3db – это набор скриптов на Perl, создающий портал для вашей музыкальной коллекции, доступный для просмотра из web. Вы можете искать и загружать треки на любое устройство в вашей сети или, еще того круче, отправить их в виде звукового потока прямо на устройства.

Да, но его установка не совсем стандартна. Придется повозиться с командной строкой (о ужас!). Вообще-то вы должны будете пристроить несколько скриптов Perl в соответствующих местах, создать для приложения базу данных и изменить файл конфигурации.

Все это предполагает, что у вас уже есть соответственно настроенный Apache (знайте, что многие дистрибутивы по умолчанию не предусматривают поддержки CGI-скриптов Perl в **httpd.conf**), и вы умеете продираться через web-сервер; но помощь наготове. Хорошее руководство по установке с домашней страницы *MyMP3db* проведет вас сквозь все эти препятствия. Одно из узких мест – зависимость от определенных модулей Perl: в вашем дистрибутиве их может не оказаться. Если у вас проблемы с запуском скриптов, это означает, что отсутствует пакет Perl вроде *MP3::Info* или любой другой из указанных в списке на странице установки. Вы легко можете стянуть их со CPAN, но настоятельно рекомендуется сначала проверить их наличие в вашем стандартном репозитории пакетов, чтобы не нарушать цельность вашей системы.

› Потоковая музыка с вашего web-сервера.



Настольный калькулятор

Speedcrunch

Версия 0.7 **Сайт** http://speedcrunch.digitalfanatics.org

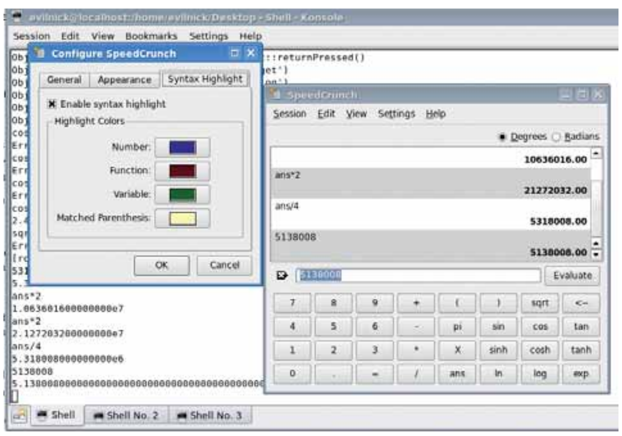
Давным-давно, в конце семидесятых – начале восьмидесятых, цифровые калькуляторы произвели мировую революцию. Вдруг, вместо того, чтобы часами суммировать большие числи на бумаге и проверять и перепроверять результат, стало возможным взять устройство размером с кирпич, ввести число 5,318,008 и вывернуть его наизнанку на потеху ученым приятелям.

Конечно же, люди до сих пор нуждаются в суммировании, и хотя последние настольные калькуляторы уже умеют строить графики, решать уравнения, а то и гулять с собакой, для действительно классного жонглирования цифрами нет ничего лучше настольного калькулятора в вашем ПК.

Speedcrunch существует уже давно, но эта новая версия добавила несколько новых полезных функций, наряду с переходом на *Qt 4*. А вам подавай, чтобы ваш калькулятор хорошо смотрелся? Но это же не главное, верно? Подсветка синтаксиса облегчает чтение результатов в груде вычислений. Калькулятор *Speedcrunch* содержит множество встроенных функций, которые можно использовать в области ввода. Правда, в основном это тригонометрические функции; если вам нужен статистический анализ, все еще не обойтись без *OpenOffice.org*!

Speedcrunch предоставляет чуть большую функциональность, чем стандартные инструменты рабочего стола, и с ним приятно поразвлечься. LXF

› Сотни человеко-часов разработки не должны пропасть даром…

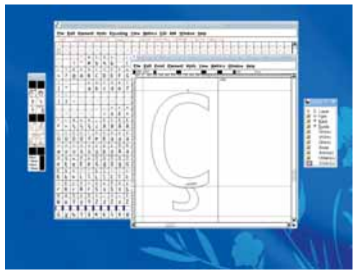


Также выпущены

Новые и обновленные программы, заслуживающие вашего внимания…

» Amsterdam Compiler Kit 6.0pre3 Переносимый компилятор и сопутствующий инструментарий для различных архитектур http://tack.sourceforge.net

» Fontforge Редактор шрифтов PostScript и TrueType http://fontforge.sourceforge.net



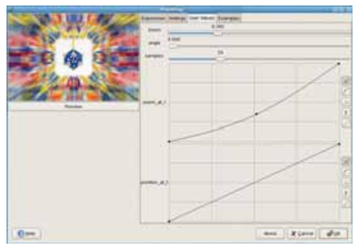
› Переделать можно почти любой шрифт.

» GnuPod 0.99.2 Инструмент, заставляющий ваш iPod работать с Linux www.gnu.org/software/gnupod

» KSquirrel 0.7 Графический просмотрщик со слайд-шоу http://ksquirrel.sourceforge.net

» Mambo 4.6.2 CMS-система http://source.mambo-foundation.org

» MathMap 1.2.0 Расширение GIMP, реализующее процедурные эффекты www.complang.tuwien.ac.at/schani/mathmap



› Изучайте причуды MathMap.

» Pybonjour 1.0 Интерфейс для системы Bonjour DNS от Apple, написанный на Python http://o2s.csail.mit.edu/o2s-wiki/pybonjour

» Python Rope IDE для Python http://rope.sourceforge.net

» Qtractor 0.0.1 Звуковая утилита http://qtractor.sourceforge.net

» RPMOrphan 1.0 Поиск «осиротевших» пакетов в вашей системе http://rpmorphan.sourceforge.net

» TribalCMS 5.0.0 Еще одна CMS-система http://tribalcms.com



LXF DVD94

Двусторонний 8-ГБ диск с четырьмя фантастическими дистрибутивами!



Майк Сондерс любовно компонует диск Linux Format, а также поддерживает сайт www.linuxformat.co.uk.

Дистро-лихорадка

DVD этого месяца так изобилует сокровищами Linux, что я воздержусь от своих обычных вступительных славословий и сразу перейду к делу – к рассказу о четырех превосходных дистрибутивах, предлагаемых вашему вниманию. На стороне 1 мы записали Mandriva 2007 Spring, обновление к одному из самых дружелюбных вариантов Linux. Если вы приобрели журнал *Linux Format* впервые и обдумываете, как легко и плавно войти в мир Linux, я настоятельно рекомендую Mandriva. В нем полным-полно замечательных программ, и он снабжен весьма полезными инструментами настройки, которые спасут вас от мороки с командной строкой.

Со стороны 2 загружаются SimplyMepis 6.5 и Ubuntu 7.04 – но это не просто стандартная версия Ubuntu. Я усовершенствовал и расширил ее сотнями дополнительных пакетов, включая KDE, *Xfce* и инструменты разработки, так что вы получите сразу Ubuntu, Kubuntu и Xubuntu – все в одном! SimplyMepis – один из самых притягательных Live-дистрибутивов: он стартует поразительно быстро, если учесть, что загрузка идет с DVD, и щеголяет усовершенствованным рабочим столом KDE. Он дает вам шанс познакомиться с Linux без его установки на жесткий диск, и это отличный способ определить совместимость вашего ПК с Linux или продемонстрировать возможности и мощь Linux друзьям и коллегам.

И, наконец, представляем вам долгожданный релиз Debian 4.0 (в виде образа CD, поскольку он не может загружаться с DVD вместе с Ubuntu). Возможно, в Debian и нет таких мастеров-помощников, как в Mandriva, или супер-свежих пакетов, как в Ubuntu, но он на редкость хорошо протестирован и надежен, что делает его отличным выбором для рабочих станций или серверов. А к этому пиршеству дистрибутивов мы приложили отборное новое программное обеспечение, материалы предыдущих выпусков в формате PDF и свыше 80 проблем с Linux, решение которых предлагается в разделе **Ответы**.

Шаг за шагом: Установка Ubuntu Linux



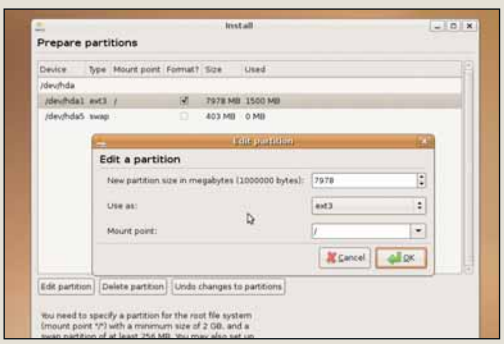
1 Загрузка

Загрузите компьютер со стороны 2 нашего DVD и нажмите Enter в меню загрузки. (Если возникнут проблемы, перегрузите компьютер в безопасном графическом режиме.)



2 Установка

Ubuntu загрузится. Войдя в рабочий стол Gnome, щелкните дважды по значку Install вверху слева.



5 Разделы

Если вы выделяете раздел на диске вручную, мы рекомендуем 10+ ГБ ext3 для корневого (/) раздела и 512 МБ на раздел подкачки.



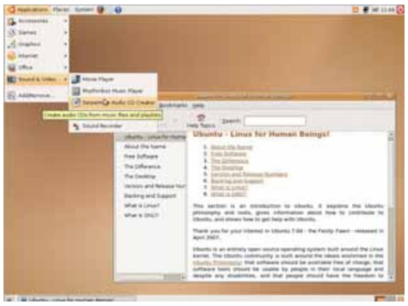
6 Переход

Если вы устанавливаете дистрибутив на машину с установленным Windows, перед вами, возможно, появится предложение скопировать некоторые настройки.

Дистрибутив Linux

Ubuntu 7.04

Если какой-либо дистрибутив Linux и сумел приблизиться к статусу стандартного, особенно на рынке домашних настольных компьютеров, то это Ubuntu. ИТ-гигант Dell планирует предустановку Ubuntu на свои машины – с такой поддержкой вполне понятно, что для многих людей Ubuntu очень скоро станет синонимом Linux. Конечно, мы все любим изучать новые дистрибутивы, но наличие одной хорошей платформы для массового рынка, без сомнения, поможет Linux расправить крылья.



› **Быстрый, надежный и с хорошей поддержкой – его популярность неудивительна.**

И вот появился Ubuntu 7.04 (по прозвищу Feisty Fawn – Отважный Олененок). Список новых функций впечатляет:

» **Инструмент перехода из Windows** Если вы устанавливаете Ubuntu на машине с двойной загрузкой Linux/Windows, эта утилита скопирует различные настройки, такие, как обои рабочего стола, закладки *Internet Explorer* и контакты AOL/YM.

» **Avahi** Сетевая система, не требующая настройки. Она позволит вам находить в сети такие ресурсы, как принтеры и файлы с совместным доступом.

» **Поддержка мультимедиа** Новая программа поможет вам реализовать поддержку несвободных форматов без ручного добавления репозиториев и выбора соответствующих пакетов.

Все эти новинки поставляются вместе с ядром 2.6.20 и Gnome 2.18. Но, как упоминалось во введении, мы добавили сотни дополнительных пакетов, и фанатам KDE или *Xfce* не придется убивать время на поиск любимого рабочего стола после установки. Просто выйдите из Gnome и измените сессию с помощью кнопки внизу слева на экране входа в систему, и все.



› **Добавление поддержки несвободных медиаформатов – пустяк для Feisty Fawn.**

Чтобы установить или обновить Ubuntu, понадобится не менее 256 МБ ОЗУ; для нормальной работы мы рекомендуем машину с процессором 800+ МГц и не менее 10 ГБ пространства на жестком диске. Если вы планируете двойную загрузку вместе с имеющейся ОС Windows (и других ОС у вас не установлено), Ubuntu может изменить размер раздела Windows. Чтобы система заработала, загрузите компьютер со стороны 2 нашего DVD и нажмите Enter в меню. Загружаемая версия на нашем DVD 32-битная, она с успехом будет работать на машинах AMD64, но если вам нужна настоящая 64-битная версия, на стороне 2 в разделе **Дистрибутивы** вас ждет ее ISO-образ. »



3 Детали

Программа установки запустится и предложит вам выбрать язык, местоположение и раскладку клавиатуры.



4 Диски

На стадии разбиения диска на разделы можно разрешить Ubuntu потеснить любой раздел Windows, ради свободного пространства на диске, а можно выделить разделы вручную.



7 Выбор имени

Создайте здесь свою учетную запись пользователя – да не забудьте пароль, который введете!



8 Поехали!

Ubuntu установится, и после того, как файлы будут скопированы, он перезагрузится – извлеките DVD и резвитесь.

Дистрибутив Linux

Mandriva 2007 Spring Free

Создание дружелюбного дистрибутива Linux отнимает много времени и усилий. Нельзя просто поставить самую свежую версию KDE поверх ядра и надеяться, что все вокруг начнут этим пользоваться – нужна хорошо продуманная программа установки, отличное распознавание оборудования и бездна инструментов настройки. Mandriva традиционно лидировала во всех этих сферах, и обновление-2007 не стало исключением, предоставляя отличную возможность познакомиться с Linux.

Spring – это возвращение к полугодовому циклу выхода релизов, после того, как Mandriva наигралась с идеей выпуска релизов раз в год. Он построен на ядре 2.6.17 с *X.org 7.1*, KDE 3.5.6, *OpenOffice.org 2.1* и *Firefox 2.0*. Это скорее эволюционный, нежели революционный релиз: основные новые функции – трехмерный оконный менеджер *Metisse* и обновленный менеджер пакетов.

Чтобы установить Mandriva, вставьте DVD в дисковод (Сторона 1) и перегрузите машину. Если ваш компьютер настроен на загрузку DVD-ROM, перед вами должен появиться экран загрузки Mandriva, точь-в-точь как на рисунке; если этого не произошло, вам придется настроить BIOS вашего ПК на загрузку с DVD, а не с жесткого диска.

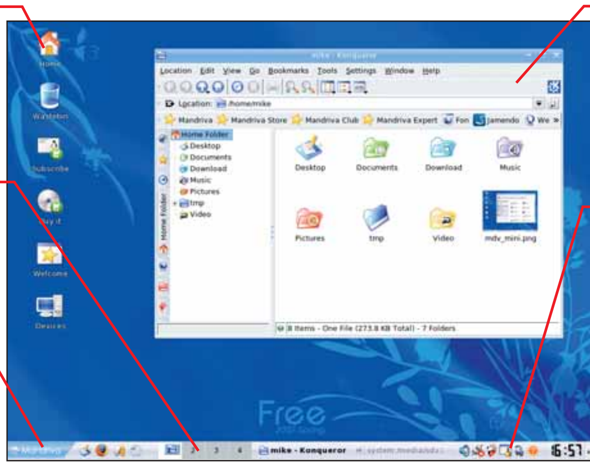
После загрузки Mandriva с DVD запустится программа установки, она задаст вам несколько вопросов. В большинстве случаев эта программа будет графической; если окажется, что она в текстовом режиме, управляйте ею с помощью курсора и клавиш Tab, пробел и Enter. Программа разобьет ваш диск на разделы – можете разрешить ей делать это автоматически, а можете задать размер разделов (частей на диске) сами. Как минимум, вам потребуется 10-ГБ корневой раздел (/), там будут храниться основные файлы, и 512-МБ раздел подкачки [swap partition], который служит виртуальной памятью. Программа установки проведет вас через весь процесс.

Если вы не уверены, что вам нужно, примите пакеты, выбранные по умолчанию (позднее вы сможете добавить еще), а когда программа загрузки предложит вам ввести пароль администратора и подробную информацию об учетных записях пользователей, не забудьте их! Помните также, что они чувствительны к регистру.

По прохождении процесса установки, Mandriva перезагрузится (извлеките DVD), и перезапустит компьютер.

Вы можете игнорировать мастер, появляющийся при первом запуске, и, войдя в систему как ранее созданный пользователь, с соответствующим паролем, без проблем наслаждаться Linux! Чтобы настроить систему, нажмите на расположенную внизу слева кнопку Mandriva > System> Configuration > Configure Your Computer. Запустится *Центр Управления Mandriva*. В случае проблем при установке или использовании Mandriva, вам помогут другие пользователи на www.linuxforum.ru.

Исследуем рабочий стол Mandriva



Рабочий стол
Home хранит ваши персональные файлы; Devices – информацию о внешних и внутренних устройствах.

Рабочее пространство
Здесь можно переключаться между виртуальными рабочими столами, это дает вам больше пространства для открытия окон.

Главное меню
Нажмите на кнопку Mandriva, и увидите, какие программы уже установлены.

Konqueror
Это – файловый менеджер для Mandriva; он же – высокоскоростной web-браузер.

Системный лоток
Некоторые апплеты, например, программы напоминания и слежения за батареей ноутбука, размещают свои значки здесь.



› Чтобы использовать *Центр Управления Mandriva*, введите свой пароль администратора.

Шаг за шагом: Установка Mandriva Linux 2007



1 Загрузка
Вставьте DVD в дисковод (сторона 1) и перезагрузите компьютер. Когда появится это меню загрузки, нажмите стрелку вниз, и затем – Enter, чтобы начать. Следуйте подсказкам, а при затруднении нажмите Help.



2 Разделы
Mandriva может поделить жесткий диск на разделы автоматически; если вы делаете это вручную, создайте раздел администратора root (/) объемом не менее 10 ГБ и раздел подкачки объемом 512 МБ.

Дистрибутив Linux

Debian GNU/Linux 4.0

Почти через два года после выхода Debian 3.1, после мучительного цикла разработки, был завершен релиз 4.0. В типичной манере Debian, политика, свары между разработчиками и истребление критических ошибок задержали 4.0 почти на четыре месяца. Ясно, что проект переживает не самые легкие времена, коли даже его основатель Иэн Мердок [Ian Murdock] критикует его за упущенные возможности (см. интервью в LXF92), и у нового руководителя проекта Сэма Хосвара [Sam Hocevar] полные руки проблем.

Зато, опять же в стиле Debian, релиз 4.0 исключительно солиден, утончен и отлично протестирован. Очень немногие дистрибутивы столь безжалостно тестируют свои пакеты, разрабатывают заплатки и вновь тестируют их, как Debian, и результат действительно налицо: ПО работает именно так, как вы и ожидали, и не забивает оставшуюся часть системы. Политика строгого документирования и организации файловой системы в Debian привели к созданию очень целостной системы, в которой легко ориентируются администраторы – это один из лучших серверных дистрибутивов.



› Наконец-то дождались: у Debian появился графический установщик!

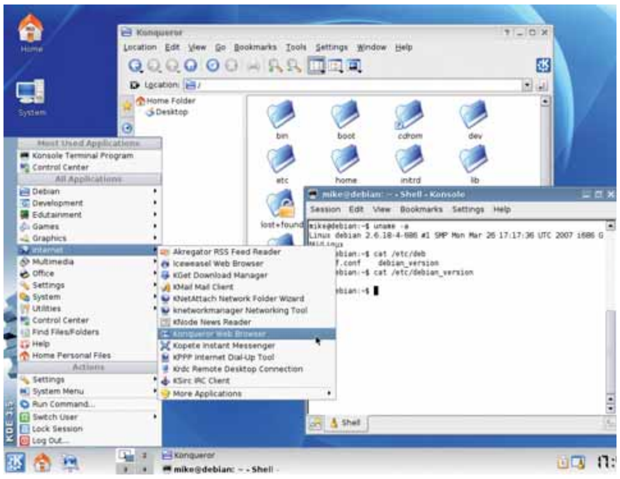
Оборотной стороной «медали за тестирование» является, естественно, тот факт, что релизы Debian не включают в себя абсолютно новых версий ПО. Те, кому подавай на рабочий стол самые свежие программы, будут огорчены, но стабильность более чем стоит этой жертвы. Релиз 4.0 базируется на ядре 2.6.18, *glibc 2.3.6* и *X.org 7.1*, с *Apache 2.2.3*, PHP 4.4.4 и *MySQL 5.0.32*, доступными для обладателей серверов LAMP.

На стороне 2 нашего DVD, в разделе Дистрибутивы/Debian, вы найдете 32-битный x86 ISO-образ CD-версии 4.0r0, который вы можете записать на CD-R и произвести с него установку (к сожалению, нам не удалось интегрировать его непосредственно в наш DVD из-за нестыковки с файлами Ubuntu). ISO-образ содержит настольную вариацию Debian, с KDE 3.5.5, *OpenOffice.org 2.0.4* и *Iceweasel 2.0.0.3* (версия *Firefox* без товарных знаков) в качестве основного ПО.

Одно из главных отличий данного релиза – новая графическая программа установки. Долгие годы текстовый установщик Debian служил дистрибутиву верой и правдой, хоть и пугал новичков, но теперь здесь имеется программа на базе *GTK*, работающая по принципу «укажи-и-кликни» (point-and-click), и она упрощает процесс, хоть и не запускается по умолчанию; вместо этого вам надо ввести в начальном загрузочном меню следующее:

```
installgui
```

Среди прочих изменений – поддержка зашифрованных разделов, более безопасная загрузка пакетов в APT и ускоренное получение списков пакетов при закачке программ из сети. Даже с новым графическим установщиком для настройки Debian придется потрудиться побольше, чем в Mandriva или Ubuntu, но награда будет достойной. »



› На нашем DVD вы найдете также полное руководство по установке (install.txt) вместе с ISO-образом, который поможет вам установить Debian.

Дистрибутив Linux

SimplyMepis 6.5

Четвертый дистрибутив на DVD этого месяца – SimplyMepis, отличный Live-дистрибутив на базе Ubuntu. SimplyMepis работает прямо с DVD, его не надо устанавливать, так что это – идеальная возможность поэкспериментировать с Linux и проверить совместимость с Linux нового компьютера. Как видно из номера версии, глобальных отличий от версии 6.0 не наблюдается – и в самом деле, некоторые из основных пакетов программ, такие, как KDE (3.5.3) и *OpenOffice.org (2.0)*, оставлены теми же, что и в релизе 6.0, по соображениям стабильности. Так что ничего ультрасовременного, но зато похвальная концентрация на надежности. Самые большие изменения в этом релизе касаются утилит настройки Mepis, выделенных в отдельную группу инструментов. Сетевая настройка усилена поддержкой WPA и автоматическим определением сети. Если вам нужны 3D-украшения на рабочем столе, к вашим услугам *Beryl*, амбициозный менеджер окон, размещающий виртуальные рабочие пространства на гранях куба и обладающий всякими эффектами – например, полупрозрачностью.

Для запуска SimplyMepis 6.5, загрузите компьютер со стороны 2 нашего DVD, и выберите SimplyMepis: Default Options. Нажмите Enter, и дистрибутив начнет загружаться. Если на этом этапе возникнут проблемы, перезагрузите компьютер и попробуйте выбрать из загрузочного меню другие опции. Например, если появляются некие графические странности, используйте режим VESA mode.



› Да здравствует хит Linux, даже не касающийся жесткого диска!

После того, как SimplyMepis загрузится, перед вами появится экран приглашения; чтобы попасть на рабочий стол, надо ввести слово 'demo' – и как имя пользователя, и как пароль. Если вам предложат ввести пароль администратора, введите 'root'. Не забудьте, что SimplyMepis можно установить на жесткий диск, щелкнув на значок программы установщика на рабочем столе – если понадобится помощь, вы найдете ее на www.mepislovers.org.



3 Готово

После установки Mandriva ваш компьютер перезагрузится. Извлеките DVD и приступайте к ознакомлению со своей новой ОС!

Другое программное обеспечение

Новые релизы

Ваши интернет-инструменты немного устарели? В наши дни браузеры и почтовые клиенты должны постоянно обновляться. Иначе вы рискуете тем, что не сможете увидеть Google Maps (и, следовательно, лишитесь репутации профи) или пострадаете от дыры в безопасности (и, следовательно, подарите свой пароль какому-нибудь взломщику-малолетке).

Чтобы предотвратить подобные ужасы, зайдите в раздел **Интернет** на стороне 2 нашего DVD, где вас ожидают два главных новых релиза Мира Сетей: *Opera 9.2* и *Thunderbird 2.0*. *Opera* заслужила уважение как молниеносный браузер, который не пожирает память подобно *Firefox* и снабжен дополнительными функциями, например, почтовым клиентом и интернет-пейджером. В новой версии есть исключительно крутая функция, именуемая Speed Dial, которая использует миниатюры web-сайтов, чтобы вы с первого взгляда увидали, что творится на ваших любимых узлах Сети.

Спасибо команде разработки *Opera* за создание широкого ассортимента двоичных пакетов для Ubuntu, Fedora, SUSE, Mandriva и Debian; если у вас другой дистрибутив, извлеките соответствующий архив **.tar.gz** и запустите **install.sh** в полученной директории.

По части почтовых клиентов, *Thunderbird* дает возможность испробовать уйму новых функций. С помощью тэгов вы сможете приписывать произвольно выбранные слова к почтовым сообщениям, что облегчает задачу по их поиску и группировке. Теперь можно осуществлять навигацию вперед и назад по журналу просмотров своих сообщений, а новые предупреждения об опасности содержат больше информации об отправителе.

Для установки *Thunderbird 2.0*, сначала удалите все ранее установленные версии с помощью менеджера пакетов, во избежание конфликтов. Скопируйте **thunderbird-2.0.0.4.tar.gz** из раздела **Интернет** нашего DVD в домашнюю директорию, затем распакуйте и запустите его:

```
tar xfvz thunderbird-2.0.0.4.tar.gz
thunderbird/thunderbird
```

При этом создастся каталог с именем **thunderbird**. Если вам нужен ярлык на рабочем столе или запись о новой версии *Thunderbird* в меню, просто укажите в качестве команды исполняемый файл **thunderbird** внутри созданной директории, и все готово к работе!

Журнал в формате PDF

Однако на нашем DVD не только программы: мы предлагаем вам также подборку материалов из предыдущих номеров журнала, которые можно прочитать в формате PDF. Через **index.html** на второй стороне DVD, зайдите в раздел **Журнал/PDFs** в левом меню, просмотрите директорию и откройте **index.html**, где находятся ссылки. Мы начали со статьи, опубликованной в LXF83: «Ubuntu навсегда», где рассматривается история популярнейшего в мире дистрибутива и анализируются те основные функции, которые и вывели его в лидеры.

Как предыдущие релизы отражаются в наших обзорах? Годится ли дистрибутив для корпоративного использования? И что, по мнению Марка Шаттлворта [Mark Shuttleworth] подвигает людей участвовать в разработке Ubuntu? Этот материал относился к Ubuntu 6.06, так что лучше вам следовать тому руководству по установке, которое приводится здесь, нежели в файле PDF.

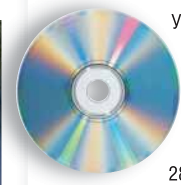


Другая статья, из LXF80, анализирует потенциальные последствия от введения новой версии General Public License (GPL) v3. Все мы пользуемся программами, распространяемыми под GPL, например, ядром Linux, GCC и всевозможными системными библиотеками, поэтому серьезные изменения в лицензии коснутся всех нас. Грэм Моррисон изучает «корни» GPL v3 и то, как она пытается справиться с возникающими сегодня вопросами патентов на ПО и DRM, а также рассказывает об опасениях в сообществе.

Завершающим материалом является опубликованное в LXF77 введение во FreeBSD, очаровательную ОС с открытым кодом, имеющую много общего с Linux, но разработанную в совершенно иной манере. FreeBSD исповедует централизованный подход, в результате чего получается крайне стабильная и хорошо документированная система. Если вам понравится FreeBSD, можете взять ISO (или произвести установку по сети) на www.freebsd.org. Наслаждайтесь исследованием ОС!

› **Узнайте больше об Ubuntu, GPL v3 и FreeBSD из статей на нашем диске.**

И напоследок...

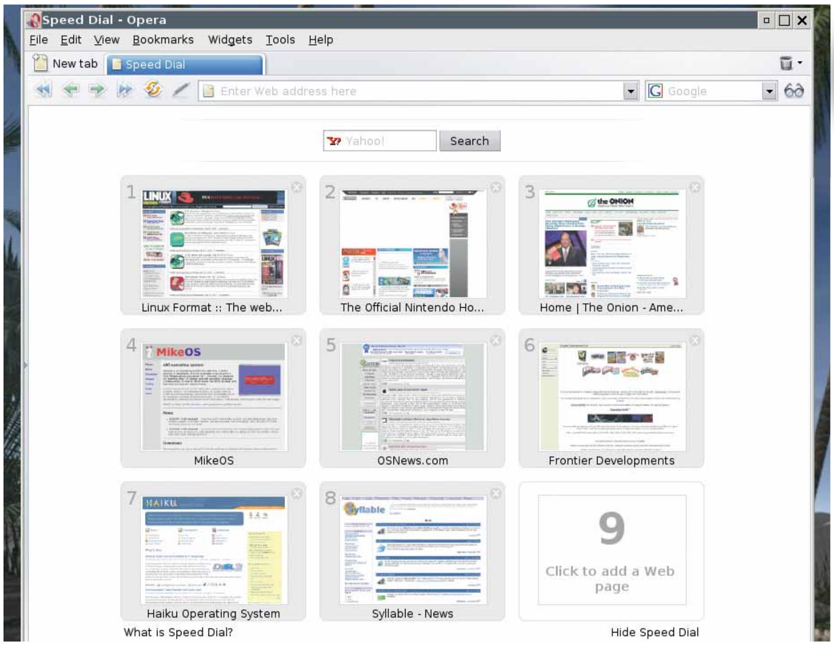


У нас есть кое-что для всех в этом месяце в разделе **Игры** на стороне 2. Поклонники скоростных гонок должны попробовать *VDrift*, быстро совершенствующийся 3D-симулятор автогонок с шикарными машинами. В данный момент *VDrift* находится в стадии активной разработки, но уже насчитывает 19 трасс и 28 машин.

Если вы ищете чего-то более значимого, то вас заинтересует *OpenTTD*, превосходный открытый клон классического *Transport Tycoon Deluxe* от Microprose. Для детей мы приготовили *Barbie Seahorse Adventures* – небольшую кокетливую платформенную игру, где вам придется летать на пузырях, чтобы добраться до луны; ну, а те, кто тоскует по славным дням 8-битной эры, должны попробовать *Briquolo*, забавную трехмерную шутку в стиле *Breakout*, в которой имеется редактор уровней. LXF



› *Briquolo*: 8-битная игра на 32-битной машине с 24-битной графикой!



› **Из-за новой функции Speed Dial в *Opera 9.2*, закладки уже выглядят вот таким Web 1.0...**

Информация о диске

ВНИМАТЕЛЬНО ПРОЧТИТЕ ЭТО ПЕРЕД ТЕМ, КАК ИСПОЛЬЗОВАТЬ DVD-ДИСК !

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, следует заглянуть именно туда.

ФОРМАТЫ ПАКЕТОВ

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любые другие. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными бинарными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы смогли собрать его самостоятельно.

ДОКУМЕНТАЦИЯ

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

ЧТО ЭТО ЗА ФАЙЛЫ?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux, различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

имя_программы-1.0.1.i386.rpm – вероятно, это бинарный пакет RPM, предназначенный для работы на системах x86;

имя_программы-1.0.1.i386.deb – такой же пакет, но уже для Debian;

имя_программы-1.0.1.tar.gz – обычно это исходный код;

имя_программы-1.0.1.tgz – тот же файл, что и выше по списку: «tgz» - это сокращение от «tar.gz»;

имя_программы-1.0.1.tar.bz2 – тот же файл, но сжатый bzip2 вместо обычного gzip;

имя_программы-1.0.1.src.rpm – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

имя_программы-1.0.1.i386.FC4.RPM – бинарный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

имя_программы-1.0.1.ppc.Suse9.rpm – бинарный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

имя_программы-devel-1.0.1.i386.rpm – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru



DVD

СОДЕРЖАНИЕ LINUX FORMAT DVD

LINUX FORMAT

ИЮЛЬ 2007

СТОРОНА 1

Mandriva – Дружелюбный дистрибутив.

СТОРОНА 2

ЖУРНАЛ

Gtk – Исходные тексты к учебнику GTK+.
JavaEE – Простая адресная книга на Java.
Mono – Код примеров статьи.
OpenOffice.org – Макросы OOo.
PDFs – Статьи из предыдущих номеров LXF.
OpenVPN – Примеры файлов настройки

РАБОЧИЙ СТОЛ

AfterStep – Next-подобный оконный менеджер.
Emulators – Эмуляторы различных архитектур.
FOSS Beyond Linux – Свободные программы для Windows и Mac.
Happydigger – Инструмент создания каталога археологических находок.
K-Letter-Wizard – Помощник в написании писем.
PeaZip – Менеджер архивов.
Referencer – Органайзер документов.
Sakura – Эмулятор терминала.
Squeeze – Менеджер архивов Xfce.

РАЗРАБОТКА

E – Язык программирования.
Mono – Среда .NET.
MonoDevelop – IDE для C#.
Qt – Кросс-платформенный GUI-инструментарий.

БЕЗОПАСНОСТЬ

EtherApe – Графический сетевой монитор.
Ethereal – Анализатор протоколов.
Logsend – Мониторинг файлов журналов.

ДИСТРИБУТИВЫ

Debian – Суперстабильный дистрибутив.
SimplyMepis – Основанный на Ubuntu LiveCD с возможностью установки.
Ubuntu – Самый популярный в мире дистрибутив.

ИГРЫ

Barbie Seahorse
Adventures – Платформенная игра.
Briquolo – Breakout-подобная игра.
OpenTTD – Клон Transport Tycoon.
Overgod – Многопользовательская 2D-стрелялка.
VDrift – Автосимулятор.

ГРАФИКА

Dia – Приложения для создания графиков и диаграмм.
Gnofract 4D – Генератор фрактальных образов.
QIV – Скоростная программа просмотра изображений.

ИНТЕРНЕТ

CGmail – Программа проверки ящика GMail.
GFTP – Основанный на GTK FTP-клиент.
Opera – Web-браузер.
Pidgin – Интернет-пейджер (бывший Gaim).
Thunderbird – Почтовый клиент Mozilla.

СПРАВКА

Answers – Решенные проблемы Linux.
Rute – Руководство по администрированию Linux.

HOTPICKS

Fracplanet – Генератор планет и земной поверхности.
Gramps – Генеалогическая программа.

KWA – Моделирование тушения пожаров.
MyMP3db – Сервер для MP3-коллекции.
Semantik – Карты разума.
SpeedCrunch – Калькулятор для рабочего стола.
Sylpheed – Почтовый клиент.
Uade – Unix Amiga Delitracker Emulator.
Viking – Менеджер данных GPS.
Warzone 2100 – Стратегия в реальном времени.

БЕЗОПАСНОСТЬ

EtherApe – Графический сетевой монитор.
Ethereal – Анализатор протоколов.
Logsend – Мониторинг файлов журналов.

СИСТЕМА

Metamorphose – Программа для массового переименования файлов.
StartCom – Обновления StartCom EL 5.
Wine – Слой совместимости с Windows.
WineTools – Оболочка к Wine в виде меню.

ГЛАВНОЕ

ATI driver – Графический драйвер.
Bash – Оболочка командной строки.
Cairo – Графическая двумерная библиотека.
CheckInstall – Программа для создания двоичных пакетов.
Coreutils – Утилиты командной строки.
GLib – Низкоуровневая библиотека.
Glibc – Библиотека GNU C.
GTK – GUI-инструментарий.
HardInfo – Системная информация и анализ.
Jigdo – Создатель ISO-образов.
Kernel – Свежий релиз ядра Linux.
LibXML2 – XML-анализатор и набор инструментов.
Ncurses – Инструментарий текстового режима.
Nvidia driver – Графический драйвер.
Python – Язык программирования.
Rawrite – Программа записи образов на дискеты.
SBM – The Smart Boot Manager.
SDL – Библиотека мультимедиа.

Комментарии? Присылайте ваши мысли и предложения по электронной почте: info@linuxformat.ru
Пожалуйста, ознакомьтесь с опубликованной в журнале инструкцией перед использованием данного диска.
Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не может принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Дефектные диски. В маловероятном случае обнаружения дефектов на данном диске, пожалуйста, обращайтесь по адресу: disks@linuxformat.ru

Тираж изготовлен на Уральском электронном заводе, 620066, Россия, г. Екатеринбург, ул. Комсомольская 17-203, Лицензия МПТР России ВАФ № 77-15

СОЗДАНИЕ УСТАНОВОЧНЫХ ДИСКОВ ПРИ ПОМОЩИ CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Для начала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем, после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь вы можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust!. Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что, если у вас нет устройства, с помощью которого можно было записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков, подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика. LXF

DVD в первом приближении

СТОРОНА 1

ДИСТРИБУТИВЫ

Mandriva — Дружелюбный дистрибутив.

СТОРОНА 2

ЖУРНАЛ

Gtk — Исходные тексты к учебнику GTK+.
JavaEE — Простая адресная книга на Java.
Mono — Код примеров статьи.
OpenOffice.org — Макросы OOo.
PDFs — Статьи из предыдущих номеров LXF.
OpenVPN — Примеры файлов настройки

РАБОЧИЙ СТОЛ

AfterStep — Next-подобный оконный менеджер.
Emulators — Эмуляторы различных архитектур.
FOSS Beyond Linux — Свободные программы для Windows и Mac.
Happydigger — Инструмент создания каталога археологических находок.
K-Letter-Wizard — Помощник в написании писем.
PeaZip — Менеджер архивов.
Referencer — Органайзер документов.
Sakura — Эмулятор терминала.
Squeeze — Менеджер архивов Xfce.

ИГРЫ

Barbie Seahorse Adventures — Платформенная игра.
Briquolo — Breakout-подобная игра.
OpenTTD — Клон Transport Tycoon.
Overgod — Многопользовательская 2D-стрелялка.
VDrift — Автосимулятор.

РАЗРАБОТКА

E — Язык программирования.
Mono — Среда .NET.
MonoDevelop — IDE для C#.
Qt — Кросс-платформенный GUI-инструментарий.

ДИСТРИБУТИВЫ

Debian — Суперстабильный дистрибутив.
SimplyMepis — Основанный на Ubuntu LiveCD с возможностью установки.
Ubuntu — Самый популярный в мире дистрибутив.

ГРАФИКА

Dia — Приложения для создания графиков и диаграмм.
Gnofract 4D — Генератор фрактальных образов.
QIV — Скоростная программа просмотра изображений.

ИНТЕРНЕТ

CGmail — Программа проверки ящика GMail.
GFTP — Основанный на GTK FTP-клиент.
Opera — Web-браузер.
Pidgin — Интернет-пейджер (бывший Gaim).
Thunderbird — Почтовый клиент Mozilla.

ИНТЕРНЕТ

CGmail — Программа проверки ящика GMail.
GFTP — Основанный на GTK FTP-клиент.
Opera — Web-браузер.
Pidgin — Интернет-пейджер (бывший Gaim).
Thunderbird — Почтовый клиент Mozilla.

СПРАВКА

Answers — Решенные проблемы Linux.
Rute — Руководство по администрированию Linux.

HOTPICKS

Fracplanet — Генератор планет и земной поверхности.
Gramps — Генеалогическая программа.
KWA — Моделирование тушения пожаров.
MyMP3db — Сервер для MP3-коллекции.
Semantik — Карты разума.
SpeedCrunch — Калькулятор для рабочего стола.
Sylpheed — Почтовый клиент.
Uade — Unix Amiga Delitracker Emulator.
Viking — Менеджер данных GPS.
Warzone 2100 — Стратегия в реальном времени.

БЕЗОПАСНОСТЬ

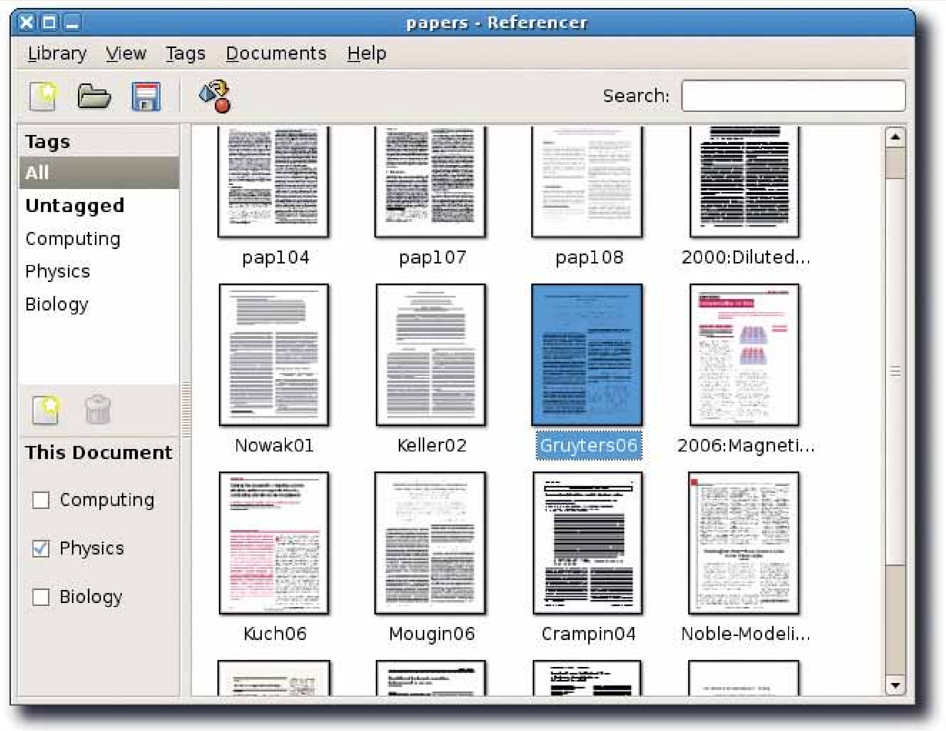
EtherApe — Графический сетевой монитор.
Ethereal — Анализатор протоколов.
Logsend — Мониторинг файлов журналов.

СИСТЕМА

Metamorphose — Программа для массового переименования файлов.
StartCom — Обновления StartCom EL 5.
Wine — Слой совместимости с Windows.
WineTools — Оболочка к Wine в виде меню.

ГЛАВНОЕ

ATI driver — Графический драйвер.
Bash — Оболочка командной строки.
Cairo — Графическая двумерная библиотека.
CheckInstall — Программа для создания двоичных пакетов.
Coreutils — Утилиты командной строки.
GLib — Низкоуровневая библиотека.
Glibc — Библиотека GNU C.
GTK — GUI-инструментарий.
HardInfo — Системная информация и сравнительный анализ.
Jigdo — Создатель ISO-образов.
Kernel — Свежий релиз ядра Linux.
LibXML2 — XML-анализатор и набор инструментов.
Ncurses — Инструментарий текстового режима.
Nvidia driver — Графический драйвер.
Python — Язык программирования.
Rawrite — Программа записи образов на дискеты.
SBM — The Smart Boot Manager.
SDL — Библиотека мультимедиа.



› **Организуйте свои документы (и ссылки между ними) с помощью *Referencer*.**





Пожарина Галина Юрьевна,
начальник отдела информационных технологий,
Санкт-Петербургский технический колледж
управления и коммерции

На чем соревнуются олимпийцы?

В то самое время, когда ведутся жаркие споры о целесообразности применения Open Source в образовании, в сообществе тренеров и участников олимпийского движения по информатике и программированию как среди школьников, так и среди студентов средних и высших учебных заведений наблюдается смена приоритетов в пользу открытого и свободного программного обеспечения.

В то самое время, когда ведутся жаркие споры о целесообразности применения Open Source в образовании, в сообществе тренеров и участников олимпийского движения по информатике и программированию как среди школьников, так и среди студентов средних и высших учебных заведений наблюдается смена приоритетов в пользу открытого и свободного программного обеспечения.



Так, предстоящая XIX Международная олимпиада по программированию в г. Загреб, Хорватия (15-22 августа 2007 года) будет проводиться с использованием следующих ресурсов:

АППАРАТНОЕ ОБЕСПЕЧЕНИЕ:

Чипсет: Intel Q965; Процессор: Core 2 Duo E4300 (1.8GHz, 2M, 800MHz FSB); ОЗУ: 512 МБ DDR2

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ:

На все компьютеры, предназначенные для проведения соревнований, будет установлена ОС Linux. Microsoft Windows предоставляться не будет. Будут установлены следующие пакеты:

- Fedora Core 6; KDE; GNOME;
- Web-браузеры: *Firefox*, *Konqueror*;
- Редакторы: *mcedit*, *joe*, *vim*, *kate*, *kwrite*, *kdevelop*, *emacs*, *xemacs, rhide, xwpe, lazarus;*
- Компиляторы: *gcc >= 4.1*, *g++ >= 4.1*, *Free Pascal >= 2.0.4* с документацией;
- Отладчики: *gdb*, *ddd*;
- Документация STL.

Для перевода заданий лидерам команд будут предоставлены компьютеры с версией Microsoft Word, обладающей многоязыковой поддержкой. Для лидеров команд, имеющих собственные ноутбуки, будет предложено Ethernet-подключение.

http://www.hsin.hr/ioi2007



Microsoft Windows покинула компьютеры участников еще в прошлом году — на Международной олимпиаде в Мексике. На компьютерах всех участников была установлена среда GNU/Linux. Перед началом соревнований участникам олимпиады была предоставлена возможность ознакомиться со своими рабочими местами и опробовать в реальном режиме работу программного обеспечения, предназначенного для проведения соревнований. С этой целью в программе олимпиады был предусмотрен двухчасовой пробный тур, участие в котором было обязательным.

Microsoft Windows покинула компьютеры участников еще в прошлом году — на Международной олимпиаде в Мексике. На компьютерах всех участников была установлена среда GNU/Linux. Перед началом соревнований участникам олимпиады была предоставлена возможность ознакомиться со своими рабочими местами и опробовать в реальном режиме работу программного обеспечения, предназначенного для проведения соревнований. С этой целью в программе олимпиады был предусмотрен двухчасовой пробный тур, участие в котором было обязательным.

На всех компьютерах участников было инсталлировано только следующее программное обеспечение:

- KDE;
- *WindowMaker*;
- *Firefox*, *Konqueror*;
- *mcedit*, *joe*, *vim*, *kate*, *kwrite*, *kdevelop*, *emacs*, *rhide*;
- *gcc-3.3*, *g++-3.3*, *Free Pascal 2.0.2* с документацией;
- *gdb*, *ddd*;
- Документация STL.

XVIII международная олимпиада по информатике проводилась в период с 13 по 20 августа 2006 года в г. Мерида. В общей сложности в олимпиаде приняли участие представители из 75 стран мира. В соревнованиях приняло участие 284 человека. Победителем олимпиады стал школьник Филип Вольски [Filip Wolski] из Польши. Второе место занял представитель Республики Корея Жо Хен Парк [Jae Hyun Park]. Замкнул тройку лучших участников олимпиады китайский школьник Фан Лонг [Fan Long]. Одиннадцатиклассник из Санкт-Петербурга Сергей Копелиович занял почетное четвертое место (**http://info.rusolymp.ru/**).

Московские студенческие олимпиады (**http://acm.msu.ru/200610/**) проводятся под управлением системы *ejudge* (**http://www.ejudge.ru/news.shtml**). С ее помощью можно просмотреть текущий статус соревнования, результаты всех участников, отослать свое решение жюри, просмотреть свои посылки и их текущий статус, просмотреть сообщения жюри, сменить пароль. С работой системы *ejudge* можно ознакомиться заранее, приняв участие в виртуальных турнирах.

Ejudge является авторским произведением А.В. Чернова. Ее разработка началась осенью 2000 года при подготовке очередной московской студенческой командной олимпиады по программированию (**http://acm.msu.ru**). Система PC2, которая использовалась ранее, уже не могла удовлетворить возросшим требованиям к проведению подобных мероприятий (в частности, она не поддерживала и до сих пор не поддерживает автоматическое выставление результата тестирования). Самая первая версия *ejudge* умела работать лишь с турнирами по системе командных чемпионатов ACM, но в дальнейшем была реализована и поддержка турниров, использующих другие правила. Сначала система применялась только для студенческих олимпиад на факультете ВМиК МГУ, а с 2002 года стала использоваться и для проведения московских школьных олимпиад по информатике (**www.informatics.ru**). С 2004 года *ejudge* используется как основная система при проведении Открытого кубка CBOSS (**www.opencup.ru**). С 2005 года *ejudge* применяется в Летней компьютерной школе. Система используется автором для поддержки некоторых учебных курсов на ВМиК МГУ. Кроме того, *ejudge* установлена в нескольких школах и учебных заведениях России, на ее основе сформирована и демо-версия ЕГЭ по информатике (**http://www.ejudge.ru/cgi-bin/new-register?contest_id=132**).

Программа *ejudge* распространяется на условиях лицензии GNU GPL. Данная лицензия дает право использовать ее произвольным образом: вносить исправления, адаптировать под свою конфигурацию, распространять (в том числе, и на коммерческой основе). По условиям лицензии GNU, любое ПО, созданное на базе исходных текстов *ejudge* (например, исправленная версия программы), является «производной работой» и должно также распространяться на условиях лицензии GNU GPL. В частности, это касается прав модификации и дальнейшего распространения системы.

Ejudge способна работать в любом более-менее современном дистрибутиве Linux. Для доступа к web-интерфейсу требуется настроенный web-сервер Apache. Система написана на языке C и состоит из совокупности демонов, отвечающих за основные сервисы, CGI-программ, реализующих web-интерфейс со стороны сервера, и утилит командной строки/утилит с текстовым интерфейсом. В настоящее время примерный объем исходного кода системы составляет порядка 150000 строк на языке C.

Потребность в подобных системах возникает и в других центрах, в которых уделяется большое внимание обучению программированию. Усилия специалистов в данной области должны быть объединены с целью автоматизации проверки широкого класса задач в области информатики.

Опыт использования компьютерного задачника в учебном процессе и при подготовке к олимпиадам позволяет утверждать, что необходимость подобных систем для рядовых преподавателей информатики и программирования невозможно переоценить. Они сделают труд преподавателя более квалифицированным, повысят эффективность и интенсивность занятий в несколько раз, позволят увеличить интерес учащихся к данному предмету.



Приведенные факты наглядно демонстрируют не отдельные попытки, а систематический и принципиально оправданный подход к организации и проведению олимпиад по информатике и программированию. Используемые средства обеспечивают прозрачность экспертизы, а также формируют достойную базу исходных текстов для подготовки к соревнованиям.

LINUX FORMAT
Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 5500 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ:
ГЛАВНЫЙ РЕДАКТОР
Валентин Синицын info@linuxformat.ru
Редактор диска
Михаил Компаниец
Литературные редакторы
Родион Водейко, Елена Толстякова, Иван Мищенко
Переводчики
Илья Аввакумов, Александр Бикмеев, Светлана Кривошеина, Александр Казанцев, Александр Кузьменков, Алексей Опарин, Валентин Развозжаев, Александр Черных, Юлия Шабунио
Допечатная подготовка
Мария Пучкова, Родион Водейко
Креативный директор
Станислав Медведев
Технический директор
Денис Филиппов
Директор по рекламе
Денис Игнатов +7 812 965 7236; +7 901 3070073
advert@linuxformat.ru
Заместитель генерального директора
Софья Виниченко
Генеральный директор
Павел Фролов
УЧРЕДИТЕЛИ
частные лица
ИЗДАТЕЛИ
Станислав Медведев, Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Текст», ООО «ППК «Текст»
188680, Ленинградская область, Всеволожский район, Колтуши, д.32
Заказ _____

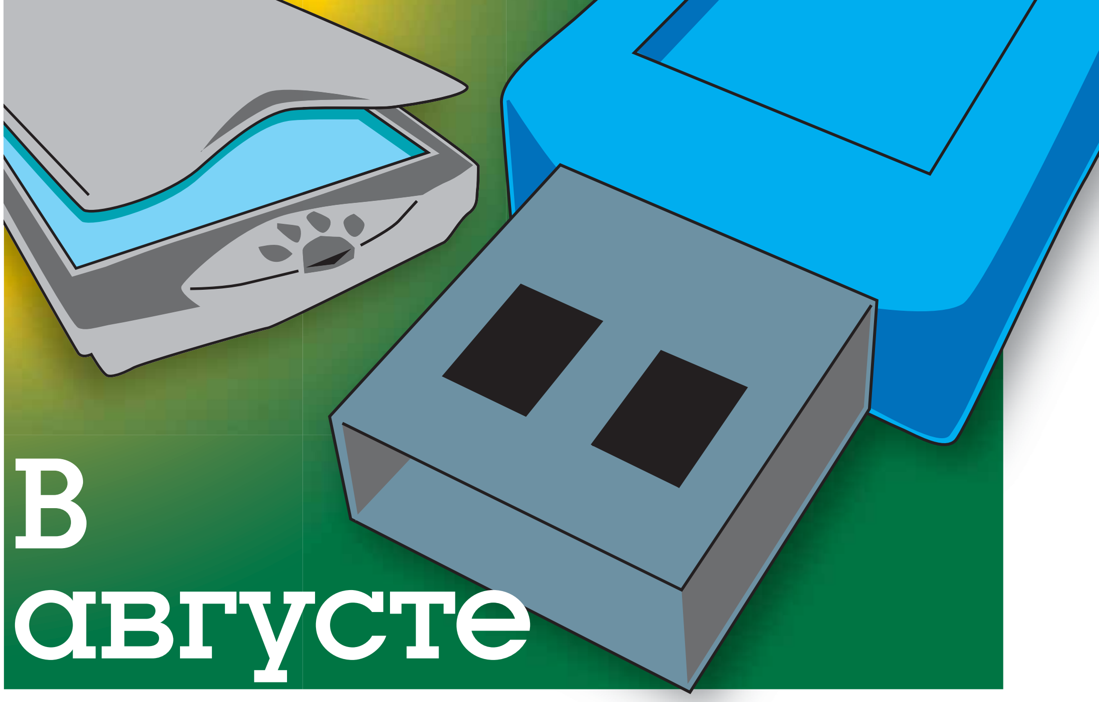
Пре–пресс: *d.r.i.v.e-group*

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ:
Редактор Ник Вейч (Nick Veitch) nick.veitch@futurenet.co.uk
Заместитель редактора Пол Хадсон (Paul Hudson) paul.hudson@futurenet.co.uk
Художественный редактор Эфрейн Эрнандес-Мендоса (Efrain Hernandez-Mendoza) efrain.hernandez-mendoza@futurenet.co.uk
Новостной редактор Майк Сондерс (Mike Saunders) mike.saunders@futurenet.co.uk
Литературный редактор
Ребекка Смолли (Rebecca Smalley) rebecca.smalley@futurenet.co.uk
Штатный автор
Грэм Моррисон (Graham Morrison) graham.morrison@futurenet.co.uk
Ассистент по выпуску
Шарлотта Мартин (Charlotte Martyn), Мэт Нейлон (Matt Nailon)

Подготовка материалов
Марк Бэйн (Mark Bain), Ладислав Боднар (Ladislav Bodnar), Нейл Ботвик (Neil Bothwick), Энди Ченнел (Andy Channelle), Дэн Фрост (Dan Frost), Маркус Грин (Marcus Green), Крис Хауэллс (Chris Howells), Энди Хадсон (Andy Hudson), Майкл Микс (Michael Meeks), Евгений Балдин, Андрей Боровский, Иван Зорин, Виктория Комиссарова, Галина Пожарина, Петр Семилетов, Александр Супрунов, Алексей Федорчук, Антон Черноусов, Илья Шпаньков, Валентин Яценко.
Художественные ассистенты: Эми Кустон (Amie Causton), Анна Фишер (Anna Fisher)
Фотографии: Джена Камбо (Jena Cumbo), Джейсон Каплан (Jason Kaplan)
Иллюстрации: Нейл Бартлет (Neil Bartlett), Пол Блехфорд (Paul Blachford), Крис Винн (Chris Winn)

КОНТАКТНАЯ ИНФОРМАЦИЯ
UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel 01225 442244 Email: linuxformat@futurenet.co.uk
РОССИЯ:
Санкт-Петербург (редакция): ул. Гончарная, 23, офис 54, телефон: (812) 717-00-37
Представительство в Москве:
пр.Мира, 161, телефон +7(495) 799-18-63, +7(495)136-88-45
Email: info@linuxformat.ru, Web: www.linuxformat.ru

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO



В августе

Железный мега-тест

Справочник покупателя: принтеры, сканеры и дюжина USB-устройств — как заставить Linux управляться со всем этим?



Раскольники от GPL3

Все мы знаем, кто хочет перейти на GPL3. А LXF отследил проекты, которые почему-то не хотят этого делать...

Сравним поддержку для предприятий

Red Hat, SUSE или Ubuntu: кто позаботится о вашей фирме лучше всех?



Интервью: Джим МакКвиллан

Все о Linux Terminal Server Project — из первых рук.

Содержание последующих выпусков может меняться без уведомления